

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-331420
(P2000-331420A)

(43) 公開日 平成12年11月30日 (2000.11.30)

(51) Int.Cl. ⁷	識別記号	F I	テマコード [*] (参考)
G 1 1 B 20/10		G 1 1 B 20/10	H
G 0 6 F 3/06	3 0 4	G 0 6 F 3/06	3 0 4 M
	12/14		3 2 0 B
G 0 9 C 1/00	6 6 0	G 0 9 C 1/00	6 6 0 D
G 1 0 L 11/00		G 1 0 L 9/00	E

審査請求 未請求 請求項の数43 O L (全 38 頁) 最終頁に続く

(21) 出願番号 特願2000-76390 (P2000-76390)

(22) 出願日 平成12年3月14日 (2000.3.14)

(31) 優先権主張番号 特願平11-69153

(32) 優先日 平成11年3月15日 (1999.3.15)

(33) 優先権主張国 日本 (J P)

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 大石 丈於

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(72) 発明者 石黒 隆二

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(72) 発明者 岡上 拓己

東京都品川区北品川6丁目7番35号 ソニ
ー株式会社内

(74) 代理人 100082762

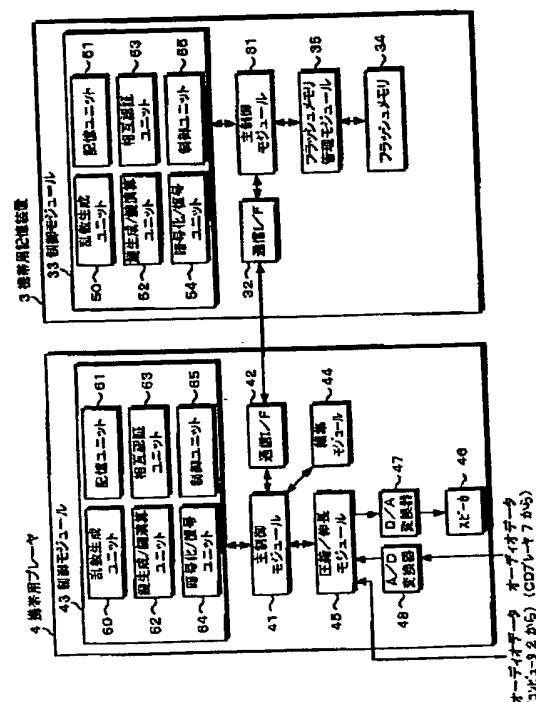
弁理士 杉浦 正知

(54) 【発明の名称】 データ処理装置、記憶装置、データ処理システムおよびその方法

(57) 【要約】

【課題】 編集処理などが行われ、既に暗号化されたトラックデータに異なる鍵データが割り当てられた場合の処理の時間を短縮する。

【解決手段】 携帯用プレーヤ4のSAM43において、コンテンツ鍵データCKとパーツ鍵データPKとのXORを演算してテンポラリ鍵データTMKを生成し、テンポラリ鍵データTMKとブロックシードデータBSとのMAC演算を行ってブロック毎にブロック鍵データBKを生成する。そして、オーディオデータをブロック毎にブロック鍵データBKを用いて暗号化してフラッシュメモリ34に記憶する。コンテンツ鍵データCKが変更になった場合には、テンポラリ鍵データTMKが変わらないように、パーツ鍵データPKを変更する。



【特許請求の範囲】

【請求項1】 単数または複数の関連するモジュールから構成されるトラックデータを、当該トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた第2の鍵データを用いて暗号化して記憶装置に出力するデータ処理装置において、
上記モジュール毎に、上記第1の鍵データおよび上記モジュールに割り当てられた上記第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、
上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて暗号化して上記記憶装置に出力する暗号化手段とを有し、
上記データ算出手段は、上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理装置。

【請求項2】 請求項1において、
上記鍵データ算出手段は、編集によって上記モジュールが、上記トラックデータに割り当てられた上記第1の鍵データとは異なる上記第1の鍵データが割り当てられた新たなトラックデータに属するようになったときに、上記モジュール毎に、上記第3の鍵データを変更しないように、上記第2の鍵データを変更するデータ処理装置。

【請求項3】 請求項1において、
上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、
上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当てられた上記第4の鍵データから第5の鍵データを算出し、
上記暗号化手段は、上記トラックデータを上記サブモジュール毎に当該サブモジュールの上記第5の鍵データを用いて暗号化して上記記憶装置に出力するデータ処理装置。

【請求項4】 請求項3において、
上記鍵データ算出手段は、上記第3の鍵データおよび上記第4の鍵データを引数とする一方向性ハッシュ関数を用いて、上記第5の鍵データを算出するデータ処理装置。

【請求項5】 請求項4において、
上記一方向性ハッシュ関数は、MAC演算を行う関数であるデータ処理装置。

【請求項6】 請求項1において、
上記鍵データ算出手段は、論理演算を行って上記第3の鍵データを算出するデータ処理装置。

【請求項7】 請求項1において、
上記データ処理装置は、
上記変更した第2の鍵データを上記記憶装置に書き込むデータ処理装置。

【請求項8】 請求項1において、

上記鍵データ算出手段は、
上記第1の鍵データおよび上記第2の鍵データを、それぞれ乱数を発生して生成するデータ処理装置。

【請求項9】 請求項8において、
上記データ処理装置は、
上記記憶装置によって暗号化された上記第1の鍵データと上記第2の鍵データとを上記記憶装置に書き込むデータ処理装置。

【請求項10】 請求項3において、
上記鍵データ算出手段は、
上記第4の鍵データを、乱数を発生して生成するデータ処理装置。

【請求項11】 請求項10において、
上記データ処理装置は、
上記生成した第4の鍵データを上記記憶装置に書き込むデータ処理装置。

【請求項12】 請求項1において、
上記記憶装置との間で相互認証を行う相互認証手段をさらに有し、
上記暗号化手段は、上記相互認証によって上記記憶装置の正当性が認められたときにのみ、上記第1の鍵データを生成するデータ処理装置。

【請求項13】 記憶装置から入力した単数または複数の関連するモジュールから構成されるトラックデータを、当該トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた第2の鍵データを用いて復号するデータ処理装置において、
上記モジュール毎に、上記第1の鍵データおよび上記モジュールに割り当てられた上記第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、
上記記憶装置から入力した上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて復号する復号手段とを有し、
上記鍵データ算出手段は、上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理装置。

【請求項14】 請求項13において、
上記鍵データ算出手段は、編集によって上記モジュールが、上記トラックデータとは異なる上記第1の鍵データが割り当てられた新たなトラックデータに属するようになったときに、上記モジュール毎に、上記第3の鍵データを変更しないように上記第2の鍵データを変更するデータ処理装置。

【請求項15】 請求項13において、
上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、
上記鍵データ算出手段は、上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当て

られた上記第4の鍵データから第5の鍵データを算出し、

上記復号手段は、上記記憶装置から入力した上記トラックデータを上記サブモジュール毎に当該サブモジュールの上記第5の鍵データを用いて復号するデータ処理装置。

【請求項16】 請求項15において、
上記鍵データ算出手段は、上記第3の鍵データおよび上記第4の鍵データを引数とする一方向性ハッシュ関数を用いて、上記第5の鍵データを算出するデータ処理装置。

【請求項17】 請求項16において、
上記一方向性ハッシュ関数は、MAC演算を行う関数であるデータ処理装置。

【請求項18】 請求項13において、
上記鍵データ算出手段は、論理演算を行って上記第3の鍵データを算出するデータ処理装置。

【請求項19】 請求項13において、
上記データ処理装置は、上記変更した第2の鍵データを上記記憶装置に書き込むデータ処理装置。

【請求項20】 請求項13において、
上記データ処理装置は、上記第1の鍵データおよび上記第2の鍵データを上記記憶装置から読み出すデータ処理装置。

【請求項21】 請求項15において、
上記データ処理装置は、上記第4の鍵データを上記記憶装置から読み出すデータ処理装置。

【請求項22】 請求項13において、
上記記憶装置との間で相互認証を行う相互認証手段をさらに有し、
上記復号手段は、上記相互認証によって上記記憶装置の正当性が認められたときにのみ、上記記憶装置から上記第1の鍵データを入力検出するデータ処理装置。

【請求項23】 記憶装置とデータ処理装置との間で、
単数または複数の関連するモジュールから構成されるトラックデータを入出力するデータ処理システムにおいて、
上記記憶装置は、記憶手段を有し、
上記データ処理装置は、上記モジュール毎に、上記トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、
上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて暗号化して上記記憶装置に出力する暗号化手段とを有し、
上記鍵データ算出手段は、上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理システム。

【請求項24】 請求項23において、

上記鍵データ算出手段は、編集によって上記モジュールが、上記トラックデータに割り当てられた上記第1の鍵データとは異なる上記第1の鍵データが割り当てられた新たなトラックデータに属するようになったときに、上記モジュール毎に、上記第3の鍵データを変更しないように上記第2の鍵データを変更するデータ処理システム。

【請求項25】 請求項23において、
上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、
上記データ処理装置は、上記鍵データ算出手段によって、上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当てられた上記第4の鍵データから第5の鍵データを算出し、
上記暗号化手段は、上記トラックデータを上記サブモジュール毎に当該サブモジュールの上記第5の鍵データを用いて暗号化して上記記憶装置に出力するデータ処理システム。

【請求項26】 請求項25において、
上記鍵データ算出手段は、上記第3の鍵データおよび上記第4の鍵データを引数とする一方向性ハッシュ関数を用いて、上記第5の鍵データを算出するデータ処理システム。

【請求項27】 請求項23において、
上記記憶装置は、第1の相互認証処理手段をさらに有し、
上記データ処理装置は、上記第1の相互認証処理手段との間で相互認証を行う第2の相互認証処理手段をさらに有するデータ処理システム。

【請求項28】 記憶装置とデータ処理装置との間で、
単数または複数の関連するモジュールから構成されるトラックデータを入出力するデータ処理システムにおいて、
上記記憶装置は、記憶手段を有し、
上記データ処理装置は、上記モジュール毎に、上記トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた上記第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、
上記記憶装置から入力した上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて復号する復号手段とを有し、
上記鍵データ算出手段は、上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理システム。

【請求項29】 請求項28において、
上記鍵データ算出手段は、編集によって上記モジュールが、上記トラックデータとは異なる上記第1の鍵データが割り当てられた新たなトラックデータに属するように

なったときに、上記モジュール毎に、上記第3の鍵データを変更しないように上記第2の鍵データを変更するデータ処理システム。

【請求項30】 請求項28において、
上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、
上記データ処理装置は、上記鍵データ算出手段によって、上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当てられた上記第4の鍵データから第5の鍵データを算出し、
上記復号手段は、上記記憶装置から入力した上記トラックデータを上記サブモジュール毎に当該サブモジュールの上記第5の鍵データを用いて復号するデータ処理システム。

【請求項31】 請求項28において、
上記記憶装置は、第1の相互認証処理手段をさらに有し、
上記データ処理装置は、上記第1の相互認証処理手段との間で相互認証を行う第2の相互認証処理手段をさらに有するデータ処理システム。

【請求項32】 単数または複数の関連するモジュールから構成されるトラックデータを、上記トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた第2の鍵データを用いて暗号化して記憶装置に記憶するデータ処理方法において、
上記モジュール毎に、上記第1の鍵データおよび上記モジュールに割り当てられた上記第2の鍵データから第3の鍵データを算出し、
上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて暗号化して上記記憶装置に出力し、
上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理方法。

【請求項33】 請求項32において、
編集によって上記モジュールが、上記トラックデータに割り当てられた上記第1の鍵データとは異なる上記第1の鍵データが割り当てられた新たなトラックデータに属するようになったときに、上記モジュール毎に、上記第3の鍵データを変更しないように、上記第2の鍵データを変更するデータ処理方法。

【請求項34】 請求項32において、
上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、
上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当てられた上記第4の鍵データから第5の鍵データを算出し、
上記トラックデータを上記サブモジュール毎に当該サブ

モジュールの上記第5の鍵データを用いて暗号化して上記記憶装置に出力するデータ処理方法。

【請求項35】 請求項34において、
上記第3の鍵データおよび上記第4の鍵データを引数とする一方向性ハッシュ関数を用いて、上記第5の鍵データを算出するデータ処理方法。

【請求項36】 記憶装置から入力した単数または複数の関連するモジュールから構成されるトラックデータを、上記トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた第2の鍵データを用いて復号するデータ処理方法において、
上記モジュール毎に、上記第1の鍵データおよび上記モジュールに割り当てられた上記第2の鍵データから第3の鍵データを算出し、
上記記憶装置から入力した上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて復号し、

上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理方法。

【請求項37】 請求項36において、
編集によって上記モジュールが、上記トラックデータとは異なる上記第1の鍵データが割り当てられた新たなトラックデータに属するようになったときに、上記モジュール毎に、上記第3の鍵データを変更しないように上記第2の鍵データを変更するデータ処理方法。

【請求項38】 請求項36において、
上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、
上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当てられた上記第4の鍵データから第5の鍵データを算出し、
上記記憶装置から入力した上記トラックデータを上記サブモジュール毎に当該サブモジュールの上記第5の鍵データを用いて復号するデータ処理方法。

【請求項39】 請求項38において、
上記第3の鍵データおよび上記第4の鍵データを引数とする一方向性ハッシュ関数を用いて、上記第5の鍵データを算出するデータ処理方法。

【請求項40】 デジタルデータを暗号化するデータ処理装置において、
コンテンツ鍵をセッション鍵で暗号化する暗号化手段と、
上記セッション鍵で暗号化されたコンテンツ鍵を記憶装置に出力するとともに、上記記憶装置の記憶用鍵データで暗号化されたコンテンツ鍵を上記記憶装置から入力するインターフェイス手段とを備え、
上記暗号化手段は、上記デジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵に基づいて暗号化

し、
上記インターフェイス手段は、上記暗号化されたデジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵とともに上記記憶装置に出力するデータ処理装置。

【請求項41】 暗号化されたデジタルデータを記憶する記憶装置において、
コンテンツ鍵をセッション鍵で復号する復号手段と、
復号されたコンテンツ鍵を記憶用鍵データで暗号化する暗号化手段と、
セッション鍵で暗号化されたコンテンツ鍵をデータ処理装置から入力し、上記記憶用鍵データで暗号化されたコンテンツ鍵を上記データ処理装置に出力し、上記暗号化されたデジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵とともに上記データ処理装置から入力するインターフェイス手段と、
上記データ処理装置から入力された暗号化されたデジタルデータとコンテンツ鍵を記憶するメモリ手段とを備える記憶装置。

【請求項42】 データ処理装置と記憶装置との間でデジタルデータを入出力するデータ処理システムにおいて、
上記データ処理装置は、
コンテンツ鍵をセッション鍵で暗号化する第1の暗号化手段と、
上記セッション鍵で暗号化されたコンテンツ鍵を上記記憶装置に出力するとともに、上記記憶装置の記憶用鍵データで暗号化されたコンテンツ鍵を上記記憶装置から入力する第1のインターフェイス手段とを備え、
上記第1の暗号化手段は、上記デジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵に基づいて暗号化し、
上記第1のインターフェイス手段は、上記暗号化されたデジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵とともに上記記憶装置に出力し、
上記記憶装置は、
上記データ処理装置からのコンテンツ鍵をセッション鍵で復号する復号手段と、
復号されたコンテンツ鍵を記憶用鍵データで暗号化する第2の暗号化手段と、
セッション鍵で暗号化されたコンテンツ鍵を上記データ処理装置から入力し、上記記憶用鍵データで暗号化されたコンテンツ鍵を上記データ処理装置に出力し、上記暗号化されたデジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵とともに上記データ処理装置から入力する第2のインターフェイス手段と、
上記データ処理装置から入力された暗号化されたデジタルデータとコンテンツ鍵を記憶するメモリ手段とを備えるデータ処理システム。

【請求項43】 データ処理装置と記憶装置との間でデ

ジタルデータを入出力するデータ処理方法において、
コンテンツ鍵をセッション鍵で暗号化するステップと、
上記セッション鍵で暗号化されたコンテンツ鍵を上記データ処理装置から上記記憶装置に送出するステップと、
上記データ処理装置からのコンテンツ鍵をセッション鍵で復号するステップと、
復号されたコンテンツ鍵を上記記憶装置の記憶用鍵データで暗号化するステップと、
記憶用鍵データで暗号化されたコンテンツ鍵を上記記憶装置から上記データ処理装置に送出するステップと、
上記デジタルデータを上記記憶装置からのコンテンツ鍵に基づいて暗号化するステップと、
上記暗号化されたデジタルデータを上記記憶用鍵データで暗号化されたコンテンツ鍵とともに上記データ処理装置から上記記憶装置に送出するステップと、
上記データ処理装置からの暗号化されたデジタルデータとコンテンツ鍵を記憶するステップとを備えるデータ処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】 この発明は、楽曲などの単数または複数の関連するモジュールから構成されるトラックデータを暗号化して記憶装置に記憶するデータ処理装置、記憶装置、データ処理システムおよびその方法に関する。

【0002】

【従来の技術】 例えば、著作権侵害となる不正利用を防止するために、1曲分のオーディオデータなどの複数の関連するモジュールから構成されるトラックデータを暗号化して記憶媒体に記憶することがある。この場合に、解読を困難にするために、例えば、個々のトラックデータに割り当てられたコンテンツ鍵データと、当該トラック内に規定されたモジュール（パーツ）に割り当てられたパーツ鍵データとを用いて演算を行ってパーツ毎にブロック鍵データを生成し、パーツ毎に異なるブロック鍵データを用いてトラックデータを暗号化する。

【0003】

【発明が解決しようとする課題】 ところで、上述したブロック鍵データを用いてトラックデータを暗号化した場合に、例えば、編集処理などが行われて、既に暗号化されたパーツが以前とは異なるコンテンツ鍵データが割り当てられた新たなトラックデータに属するようになった場合に、当該コンテンツ鍵データの変更に応じて、当該パーツに用いられるブロック鍵データが変わってしまう。その結果、当該パーツを復号した後に、再度、新たなブロック鍵データを用いて暗号化する必要が生じる。しかしながら、編集処理を行う度に、既に暗号化したトラックデータの復号および再暗号化を行うのでは、編集処理の負荷が重くなり、編集時間が長くなるという問題がある。

【0004】この発明の目的は、上述した従来技術の問題点に鑑みてなされ、編集処理などが行われ、既に暗号化されたトラックデータに異なる鍵データが割り当てられた場合の処理の時間を短縮できるデータ処理装置、記憶装置、データ処理システムおよびその方法を提供することにある。

【0005】

【課題を解決するための手段】上述した課題を解決するために、請求項1の発明は、単数または複数の関連するモジュールから構成されるトラックデータを、当該トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データを用いて暗号化して記憶装置に出力するデータ処理装置において、モジュール毎に、第1の鍵データおよびモジュールに割り当てられた第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、トラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて暗号化して記憶装置に出力する暗号化手段とを有し、データ算出手段は、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理装置である。

【0006】請求項1のデータ処理装置では、第1の鍵データが変更された場合でも、鍵データ処理手段によって第3の鍵データを変更しないように第2の鍵データが変更されるため、記憶装置に記憶した暗号化されたトラックデータを復号して再度暗号化する必要がなくなる。

【0007】請求項13の発明は、記憶装置から入力した単数または複数の関連するモジュールから構成されるトラックデータを、当該トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データを用いて復号するデータ処理装置において、モジュール毎に、第1の鍵データおよびモジュールに割り当てられた第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、記憶装置から入力したトラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて復号する復号手段とを有し、鍵データ算出手段は、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理装置である。

【0008】請求項23の発明は、記憶装置とデータ処理装置との間で、単数または複数の関連するモジュールから構成されるトラックデータを入出力するデータ処理システムにおいて、記憶装置は、記憶手段を有し、データ処理装置は、モジュール毎に、トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、トラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて暗号化して記憶装置に出力する暗号化手段とを有し、鍵データ算出手段は、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理システムである。

【0009】請求項28の発明は、記憶装置とデータ処理装置との間で、単数または複数の関連するモジュールから構成されるトラックデータを入出力するデータ処理システムにおいて、記憶装置は、記憶手段を有し、データ処理装置は、モジュール毎に、トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、記憶装置から入力したトラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて復号する復号手段とを有し、鍵データ算出手段は、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理システムである。

【0010】請求項32の発明は、単数または複数の関連するモジュールから構成されるトラックデータを、トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データを用いて暗号化して記憶装置に記憶するデータ処理方法において、モジュール毎に、第1の鍵データおよびモジュールに割り当てられた第2の鍵データから第3の鍵データを算出し、トラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて暗号化して記憶装置に出力し、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理方法である。

【0011】請求項36の発明は、記憶装置から入力した単数または複数の関連するモジュールから構成されるトラックデータを、トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データを用いて復号するデータ処理方法において、モジュール毎に、第1の鍵データおよびモジュールに割り当てられた第2の鍵データから第3の鍵データを算出し、記憶装置から入力したトラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて復号し、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理方法である。

【0012】請求項40の発明は、デジタルデータを暗号化するデータ処理装置において、コンテンツ鍵をセッション鍵で暗号化する暗号化手段と、セッション鍵で暗号化されたコンテンツ鍵を記憶装置に出力するとともに、記憶装置の記憶用鍵データで暗号化されたコンテンツ鍵を記憶装置から入力するインターフェイス手段とを備え、暗号化手段は、デジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵に基づいて暗号化し、インターフェイス手段は、暗号化されたデジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵とともに

記憶装置に出力するデータ処理装置である。

【0013】請求項41の発明は、暗号化されたデジタルデータを記憶する記憶装置において、コンテンツ鍵をセッション鍵で復号する復号手段と、復号されたコンテンツ鍵を記憶用鍵データで暗号化する暗号化手段と、セッション鍵で暗号化されたコンテンツ鍵をデータ処理装置から入力し、記憶用鍵データで暗号化されたコンテンツ鍵をデータ処理装置に出力し、暗号化されたデジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵とともにデータ処理装置から入力するインターフェイス手段と、データ処理装置から入力された暗号化されたデジタルデータとコンテンツ鍵を記憶するメモリ手段とを備える記憶装置である。

【0014】請求項42の発明は、データ処理装置と記憶装置との間でデジタルデータを入出力するデータ処理システムにおいて、データ処理装置は、コンテンツ鍵をセッション鍵で暗号化する第1の暗号化手段と、セッション鍵で暗号化されたコンテンツ鍵を記憶装置に出力するとともに、記憶装置の記憶用鍵データで暗号化されたコンテンツ鍵を記憶装置から入力する第1のインターフェイス手段とを備え、第1の暗号化手段は、デジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵に基づいて暗号化し、第1のインターフェイス手段は、暗号化されたデジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵とともに記憶装置に出力し、記憶装置は、データ処理装置からのコンテンツ鍵をセッション鍵で復号する復号手段と、復号されたコンテンツ鍵を記憶用鍵データで暗号化する第2の暗号化手段と、セッション鍵で暗号化されたコンテンツ鍵をデータ処理装置から入力し、記憶用鍵データで暗号化されたコンテンツ鍵をデータ処理装置に出力し、暗号化されたデジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵とともにデータ処理装置から入力する第2のインターフェイス手段と、データ処理装置から入力された暗号化されたデジタルデータとコンテンツ鍵を記憶するメモリ手段とを備えるデータ処理システムである。

【0015】請求項43の発明は、データ処理装置と記憶装置との間でデジタルデータを入出力するデータ処理方法において、コンテンツ鍵をセッション鍵で暗号化するステップと、セッション鍵で暗号化されたコンテンツ鍵をデータ処理装置から記憶装置に送出するステップと、データ処理装置からのコンテンツ鍵をセッション鍵で復号するステップと、復号されたコンテンツ鍵を記憶装置の記憶用鍵データで暗号化するステップと、記憶用鍵データで暗号化されたコンテンツ鍵を記憶装置からデータ処理装置に送出するステップと、デジタルデータを記憶装置からのコンテンツ鍵に基づいて暗号化するステップと、暗号化されたデジタルデータを記憶用鍵データで暗号化されたコンテンツ鍵とともにデータ処理装置から記憶装置に送出するステップと、データ処理装置

からの暗号化されたデジタルデータとコンテンツ鍵を記憶するステップとを備えるデータ処理方法である。

【0016】

【発明の実施の形態】以下、この発明の実施形態に係わるオーディオシステムについて説明する。図1は、一実施形態のオーディオシステム1のシステム構成図、図2は図1に示す携帯用記憶装置3および携帯用プレーヤ4の内部構成図である。図1に示すように、オーディオシステム1は、例えば、コンピュータ2、携帯用記憶装置3、携帯用プレーヤ4、CD-ROMドライブ6およびCDプレーヤ7を有する。

【0017】コンピュータ2

コンピュータ2は、ネットワーク5に接続されており、例えば、EMD(Electronic Music Distribution: 電子音楽配信)などのサービスを提供する図示しないサービスプロバイダのホストコンピュータから、ネットワーク5を介してオーディオデータ(トラックデータ)を受信し、当該受信したオーディオデータを必要に応じて復号して、携帯用プレーヤ4に出力する。また、コンピュータ2は、コンテンツデータを受信するに当たって、必要に応じて、サービスプロバイダのホストコンピュータとの間で認証処理および課金処理などを行う。また、コンピュータ2は、例えば、CD-ROMドライブ6から入力したオーディオデータを携帯用プレーヤ4に出力する。

【0018】携帯用記憶装置3

携帯用記憶装置3は、携帯用プレーヤ4に対して着脱自在とされ、例えば、メモリスティック(Memory Stick: 商標)であり、フラッシュメモリなどの書き換え可能な半導体メモリを内蔵している。本明細書において、メモリカードの用語が使用されることもあるが、メモリカードは、携帯用記憶装置を指すものとして使用している。図2に示すように、携帯用記憶装置3は、例えば、主制御モジュール31、通信インターフェイス32、制御モジュール33、フラッシュメモリ34およびフラッシュメモリ管理モジュール35を有する。

【0019】〔制御モジュール33〕図2に示すように、制御モジュール33は、例えば、乱数発生ユニット50、記憶ユニット51、鍵生成/演算ユニット52、相互認証ユニット53、暗号化/復号ユニット54および制御ユニット55を有する。制御モジュール33は、シングルチップの暗号処理専用の集積回路であり、多層構造を有し、内部のメモリセルはアルミニウム層などのダミー層に挟まれている。また、制御モジュール33は、動作電圧または動作周波数の幅が狭く、外部から不正にデータを読み出せないように耐タンパ性を有している。乱数発生ユニット50は、乱数発生指示を受けると、64ビット(8バイト)の乱数を発生する。

【0020】記憶ユニット51は、例えば、EEPROM(Electrically Erasable Programmable Read Only Me

memory)などの不揮発性メモリであり、認証処理に必要な鍵データなどの種々のデータを記憶している。図3は、記憶ユニット51に記憶されているデータを説明するための図である。図3に示すように、記憶ユニット51は、認証鍵データ $IK_0 \sim IK_{31}$ 、装置識別データ ID_m および記憶用鍵データ SK_m を記憶している。

【0021】認証鍵データ $IK_0 \sim IK_{31}$ は、携帯用記憶装置3が携帯用プレーヤ4との間で相互認証を行う際に用いられる鍵データであり、後述するように相互認証を行う度に認証鍵データ $IK_0 \sim IK_{31}$ のうちの認証鍵データがランダムに選択される。なお、認証鍵データ $IK_0 \sim IK_{31}$ および記憶用鍵データ SK_m は、携帯用記憶装置3の外部から読めないようになっている。装置識別データ ID_m は、携帯用記憶装置3に対してユニークに付けられた識別データであり、後述するように、携帯用記憶装置3が携帯用プレーヤ4との間で相互認証を行う際に読み出されて携帯用プレーヤ4に出力される。記憶用鍵データ SK_m は、後述するように、コンテンツ鍵データ CK を暗号化してフラッシュメモリ34に記憶する際に用いられる。

【0022】鍵生成/演算ユニット52は、例えば、ISO/IEC 9797のMAC(Message Authentication Code)演算などの種々の演算を行って鍵データを生成する。このとき、MAC演算には、例えば、“Block cipher Algorithm”としてFIPS PUB 46-2に規定されるDES(Data Encryption Standard)が用いられる。MAC演算は、任意の長さのデータを固定の長さに圧縮する一方方向性ハッシュ関数演算であり、関数値が秘密鍵に依存して定まる。

【0023】相互認証ユニット53は、携帯用プレーヤ4からオーディオデータを入力してフラッシュメモリ34に書き込む動作を行うのに先立って、携帯用プレーヤ4との間で相互認証処理を行う。また、相互認証ユニット53は、フラッシュメモリ34からオーディオデータを読み出して携帯用プレーヤ4に出力する動作を行うのに先立って、携帯用プレーヤ4との間で相互認証処理を行う。また、相互認証ユニット53は、相互認証処理において、前述したMAC演算を行う。当該相互認証処理では、記憶ユニット51に記憶されているデータが用いられる。

【0024】暗号化/復号ユニット54は、DES、IDEA、MISTYなどのブロック暗号アルゴリズムでの暗号化を行う。使用するモードは、FIPS PUB 81“DES MODES OF OPERATION”に規定されているようなECB(Electronic Code Book)モードおよびCBC(Cipher Block Chaining)モードである。また、暗号化/復号ユニット54は、DES、IDEA、MISTYなどのブロック復号アルゴリズムでの復号を行う。使用するモードは、上記ECBモードおよびCBCモードである。当該ECBモードおよ

びCBCモードのブロック暗号化/復号では、指定された鍵データを用いて指定されたデータを暗号化/復号する。制御ユニット55は、乱数発生ユニット50、記憶ユニット51、鍵生成/演算ユニット52、相互認証ユニット53および暗号化/復号ユニット54の処理を統括して制御する。

【0025】【フラッシュメモリ34】フラッシュメモリ34は、例えば、32Mバイトの記憶容量を有する。フラッシュメモリ34には、相互認証ユニット53による相互認証処理によって正当な相手であると認められたときに、携帯用プレーヤ4から入力したオーディオデータが書き込まれる。また、フラッシュメモリ34からは、相互認証ユニット53による相互認証処理によって正当な相手であると認められたときに、オーディオデータが読み出されて携帯用プレーヤ4に出力される。

【0026】以下、フラッシュメモリ34に記憶されるデータおよびそのフォーマットについて説明する。図4は、フラッシュメモリ34に記憶されるデータを説明するための図である。図4に示すように、フラッシュメモリ34には、例えば、再生管理ファイル100、トラックデータファイル1010, 1011, 1012, 1013が記憶されている。ここで、再生管理ファイル100はトラックデータファイル1010~1013の再生を管理する管理データを有し、トラックデータファイル1010~1013はそれぞれ対応するトラックデータ(オーディオデータ)を有している。なお、本実施形態では、トラックデータは、例えば、1曲分のオーディオデータを意味する。

【0027】図5は、再生管理ファイルの構成を示し、図6が一つ(1曲)のATRAC3データファイルの構成を示す。再生管理ファイルは、16KB固定長のファイルである。ATRAC3データファイルは、曲単位でもって、先頭の属性ヘッダと、それに続く実際の暗号化された音楽データとからなる。属性ヘッダも16KB固定長とされ、再生管理ファイルと類似した構成を有する。

【0028】再生管理ファイルは、ヘッダ、1バイトコードのメモリカードの名前NM1-S、2バイトコードのメモリカードの名前NM2-S、曲順の再生テーブルTRKTBL、メモリカード全体の付加情報INF-Sとからなる。データファイルの先頭の属性ヘッダは、ヘッダ、1バイトコードの曲名NM1、2バイトコードの曲名NM2、トラックのキー情報等のトラック情報TRKINF、パーツ情報PRTINFと、トラックの付加情報INFとからなる。ヘッダには、総パーツ数、名前の属性、付加情報のサイズの情報等が含まれる。

【0029】属性ヘッダに対してATRAC3の音楽データが続く。音楽データは、16KBのブロック毎に区切られ、各ブロックの先頭にヘッダが付加されている。ヘッダには、暗号を復号するための初期値が含まれる。

なお、暗号化の処理を受けるのは、ATRAC3データファイル中の音楽データのみであって、それ以外の再生管理ファイル、ヘッダ等のデータは、暗号化されない。

【0030】図7は、再生管理ファイルPBLISTのより詳細なデータ構成を示し、図8A、図8Bは、再生管理ファイルPBLISTを構成するヘッダとそれ以外の部分をそれぞれ示す。再生管理ファイルPBLISTは、1クラスタ（1ブロック＝16KB）のサイズである。ヘッダ（図8A）が32バイトである。ヘッダ以外の部分（図8B）がメモ리카ード全体に対する名前NM1-S（256バイト）、名前NM2-S（512バイト）、CONTENTS KEY、MAC、S-YMDhmsと、再生順番を管理するテーブルTRKTBL（800バイト）と、メモ리카ード全体に対する付加情報INF-S（14720バイト）であり、最後にヘッダ中の情報の一部が再度記録される。これらの異なる種類のデータ群のそれぞれの先頭は、再生管理ファイル内で所定の位置となるように規定されている。

【0031】再生管理ファイルは、（0x0000）および（0x0010）で表される先頭から32バイト（図8A）がヘッダである。なお、ファイル中で先頭から16バイト単位で区切られた単位をスロットと称する。ファイルの第1および第2のスロットに配されるヘッダには、下記の意味、機能、値を持つデータが先頭から順に配される。なお、Reservedと表記されているデータは、未定義のデータを表している。通常ヌル（0x00）が書かれるが、何が書かれていてもReservedのデータが無視される。将来のバージョンでは、変更がありうる。また、この部分への書き込みは禁止する。Optionと書かれた部分も使用しない場合は、全てReservedと同じ扱いとされる。

【0032】BLKID-TL0（4バイト）

意味：BLOCKID FILE ID

機能：再生管理ファイルの先頭であることを識別するための値

値：固定値＝"TL=0"（例えば0x544C2D30）

MCode（2バイト）

意味：MAKER CODE

機能：記録した機器の、メーカー、モデルを識別するコード

値：上位10ビット（メーカーコード） 下位6ビット（機種コード）

REVISION（4バイト）

意味：PBLISTの書き換え回数

機能：再生管理ファイルを書き換える度にインクリメント

値：0より始まり+1ずつ増加する

S-YMDhms（4バイト）（Option）

意味：信頼できる時計を持つ機器で記録した年・月・日

・時・分・秒

機能：最終記録日時を識別するための値

値：25～31ビット 年 0～99（1980～2079）

21～24ビット 月 0～12

16～20ビット 日 0～31

11～15ビット 時 0～23

05～10ビット 分 0～59

00～04ビット 秒 0～29（2秒単位）。

【0033】SN1C+L（2バイト）

意味：NM1-S領域に書かれるメモ리카ードの名前（1バイト）の属性を表す

機能：使用する文字コードと言語コードを各1バイトで表す

値：文字コード（C）は上位1バイトで下記のように文字を区別する

00：文字コードは設定しない。単なる2進数として扱うこと

01：ASCII 02：ASCII+KANA 03：modified8859-1

81：MS-JIS 82：KS C 5601-1989 83：GB2312-80 90：S-JIS(for Voice)。

【0034】言語コード（L）は下位1バイトで下記のようにEBU Tech 3258 規定に準じて言語を区別する

00：設定しない 08：German 09：English 0A：Spanish

0F：French 15：Italian 1D：Dutch

65：Korean 69：Japanese 75：Chinese

データが無い場合オールゼロとすること。

【0035】SN2C+L（2バイト）

意味：NM2-S領域に書かれるメモ리카ードの名前（2バイト）の属性を表す

機能：使用する文字コードと言語コードを各1バイトで表す

値：上述したSN1C+Lと同一

SINFSIZE（2バイト）

意味：INF-S領域に書かれるメモ리카ード全体に関する付加情報の全てを合計したサイズを表す

機能：データサイズを16バイト単位の大きさに記述、無い場合は必ずオールゼロとすること

値：サイズは0x0001から0x39C（924）

T-TRK（2バイト）

意味：TOTAL TRACK NUMBER

機能：総トラック数

値：1から0x0190（最大400トラック）、データが無い場合はオールゼロとすること

VerNo（2バイト）

意味：フォーマットのバージョン番号

機能：上位がメジャーバージョン番号、下位がマイナーバージョン番号

値:例 0x0100 (Ver1. 0)
0x0203 (Ver2. 3)。

【0036】上述したヘッダに続く領域に書かれるデータ (図8B) について以下に説明する。

【0037】NM1-S

意味: メモリカード全体に関する1バイトの名前

機能: 1バイトの文字コードで表した可変長の名前データ (最大で256)

名前データの終了は、必ず終端コード (0x00) を書き込むこと

サイズはこの終端コードから計算すること、データの無い場合は少なくとも先頭 (0x0020) からヌル (0x00) を1バイト以上記録すること

値: 各種文字コード

NM2-S

意味: メモリカード全体に関する2バイトの名前

機能: 2バイトの文字コードで表した可変長の名前データ (最大で512)

名前データの終了は、必ず終端コード (0x00) を書き込むこと

サイズはこの終端コードから計算すること、データの無い場合は少なくとも先頭 (0x0120) からヌル (0x00) を2バイト以上記録すること

値: 各種文字コード。

【0038】CONTENTS KEY

意味: 曲ごとに用意された値でMG (M) で保護されてから保存される。ここでは、1曲目に付けられるCONTENTS KEYと同じ値

機能: S-YMDhmsのMACの計算に必要となる鍵となる

値: 0から0xFFFFFFFFFFFFFFFFFまでMAC

意味: 著作権情報改ざんチェック値

機能: S-YMDhmsの内容とCONTENTS KEYから作成される値

値: 0から0xFFFFFFFFFFFFFFFFFまで。

【0039】TRK-nnn

意味: 再生するATRAC3データファイルのSQN (シーケンス) 番号

機能: TRKINFの中のFNoを記述する

値: 1から400 (0x190)

トラックが存在しない時はオールゼロとすること

INF-S

意味: メモリカード全体に関する付加情報データ (例えば写真、歌詞、解説等の情報)

機能: ヘッダを伴った可変長の付加情報データ
複数の異なる付加情報が並べられることがある。それぞれにIDとデータサイズが付けられている。個々のヘッダを含む付加情報データは最小16バイト以上で4バ

イトの整数倍の単位で構成される。その詳細については、後述する

値: 付加情報データ構成を参照

S-YMDhms (4バイト) (Option)

意味: 信頼できる時計を持つ機器で記録した年・月・日・時・分・秒

機能: 最終記録日時を識別するための値、EMDの時は必須

値: 25~31ビット 年 0~99 (1980~2079)

21~24ビット 月 0~12

16~20ビット 日 0~31

11~15ビット 時 0~23

05~10ビット 分 0~59

00~04ビット 秒 0~29 (2秒単位)。

【0040】再生管理ファイルの最後のスロットとして、ヘッダ内のものと同一のBLKID-TL0と、M Codeと、REVISIONとが書かれる。

【0041】民生用オーディオ機器として、メモリカードが記録中に抜かれたり、電源が切れることがあり、復活した時にこれらの異常の発生を検出することが必要とされる。上述したように、REVISIONをブロックの先頭と末尾に書き込み、この値を書き換える度に+1インクリメントするようにしている。若し、ブロックの途中で異常終了が発生すると、先頭と末尾のREVISIONの値が一致せず、異常終了を検出することができる。REVISIONが2個存在するので、高い確率で異常終了を検出することができる。異常終了の検出時には、エラーメッセージの表示等の警告が発生する。

【0042】また、1ブロック (16KB) の先頭部分に固定値BLKID-TL0を挿入しているので、FATが壊れた場合の修復の目安に固定値を使用できる。すなわち、各ブロックの先頭の固定値を見れば、ファイルの種類を判別することが可能である。しかも、この固定値BLKID-TL0は、ブロックのヘッダおよびブロックの終端部分に二重に記述するので、その信頼性のチェックを行うことができる。なお、再生管理ファイルBLISTの同一のものを二重に記録しても良い。

【0043】ATRAC3データファイルは、トラック情報管理ファイルと比較して、相当大きなデータ量 (例えば数千のブロックが繋がる場合もある) であり、ATRAC3データファイルに関しては、後述するように、ブロック番号BLOCK SERIALが付けられている。但し、ATRAC3データファイルは、通常複数のファイルがメモリカード上に存在するので、CONNUM0でコンテンツの区別を付けた上で、BLOCK SERIALを付けないと、重複が発生し、FATが壊れた場合のファイルの復旧が困難となる。

【0044】同様に、FATの破壊までにはいたらないが、論理を間違えてファイルとして不都合のあるような

場合に、書き込んだメーカーの機種が特定できるように、メーカーコード (MC o d e) がブロックの先頭と末尾に記録されている。

【0045】図8Cは、付加情報データの構成を示す。付加情報の先頭に下記のヘッダが書かれる。ヘッダ以降に変長のデータが書かれる。

【0046】INF

意味: FIELD ID

機能: 付加情報データの先頭を示す固定値

値: 0x69

ID

意味: 付加情報キーコード

機能: 付加情報の分類を示す

値: 0から0xFF

SIZE

意味: 個別の付加情報の大きさ

機能: データサイズは自由であるが、必ず4バイトの整数倍でなければならない。また、最小16バイト以上のこと。データの終わりより余りがでる場合はヌル (0x00) で埋めておくこと

値: 16から14784 (0x39C0)

MC o d e

意味: MAKER CODE

機能: 記録した機器の、メーカー、モデルを識別するコード

値: 上位10ビット (メーカーコード) 下位6ビット (機種コード)

C+L

意味: 先頭から12バイト目からのデータ領域に書かれる文字の属性を表す

機能: 使用する文字コードと言語コードを各1バイトで表す

値: 前述のSN1C+Lと同じ

DATA

意味: 個別の付加情報データ

機能: 可変長データで表す。実データの先頭は常に12バイト目より始まり、長さ (サイズ) は最小4バイト以上、常に4バイトの整数倍でなければならない。データの最後から余りがある場合はヌル (0x00) で埋めること

値: 内容により個別に定義される。

【0047】以下、トラックデータファイル1010～1013について説明する。図9は、トラックデータファイル1010の構成を説明するための図である。図9に示すように、トラックデータファイル1010は、1個のパーツからなり、当該パーツが5個のクラスタCL (0), CL (1), CL (2), CL (3), CL (4) で構成されている。当該パーツは、クラスタCL (0) の先頭から開始し、クラスタCL (4) のサウンドユニットSU (4) で終了している。なお、トラック

データファイル1011～1013は、基本的に、図9に示す構成をしているが、パーツ数、クラスタ数およびクラスタ内に含まれるサウンドユニットSUの数は、図9に示すものには限定されず、独立して決められている。

【0048】1トラックは、1曲を意味する。1曲は、1つのATRAC3データファイル (図6参照) で構成される。ATRAC3データファイルは、ATRAC3により圧縮されたオーディオデータである。メモ리카ード40に対しては、クラスタと呼ばれる単位で記録される。1クラスタは、例えば16KBの容量である。1クラスタに複数のファイルが混じることがない。フラッシュメモリ42を消去する時の最小単位が1ブロックである。音楽データを記録するのに使用するメモ리카ード40の場合、ブロックとクラスタは、同意語であり、且つ1クラスタ=1セクタと定義されている。

【0049】1曲は、基本的に1パーツで構成されるが、編集が行われると、複数のパーツから1曲が構成されることがある。パーツは、録音開始からその停止までの連続した時間内で記録されたデータの単位を意味し、通常は、1トラックが1パーツで構成される。曲内のパーツのつながりは、各曲の属性ヘッダ内のパーツ情報PRTINFで管理する。すなわち、パーツサイズは、PRTINFの中のパーツサイズPRTSIZEという4バイトのデータで表す。パーツサイズPRTSIZEの先頭の2バイトがパーツが持つクラスタの総数を示し、続く各1バイトが先頭および末尾のクラスタ内の開始サウンドユニット (SUと略記する) の位置、終了SUの位置を示す。このようなパーツの記述方法を持つことによって、音楽データを編集する際に通常、必要とされる大量の音楽データの移動をなくすることが可能となる。ブロック単位の編集に限定すれば、同様に音楽データの移動を回避できるが、ブロック単位は、SU単位に比して編集単位が大きすぎる。

【0050】SUは、パーツの最小単位であり、且つATRAC3でオーディオデータを圧縮する時の最小のデータ単位である。44.1kHzのサンプリング周波数で得られた1024サンプル分 (1024×16ビット×2チャンネル) のオーディオデータを約1/10に圧縮した数百バイトのデータがSUである。1SUは、時間に換算して約23m秒になる。通常は、数千に及ぶSUによって1つのパーツが構成される。1クラスタが42個のSUで構成される場合、1クラスタで約1秒の音を表すことができる。1つのトラックを構成するパーツの数は、付加情報サイズに影響される。パーツ数は、1ブロックの中からヘッダや曲名、付加情報データ等を除いた数で決まるために、付加情報が全く無い状態が最大数 (645個) のパーツを使用できる条件となる。

【0051】図10は、1SUがNバイト (例えばN=384バイト) の場合のATRAC3データファイルA

3Dnnnnのデータ配列を示す。図10には、データファイルの属性ヘッダ(1ブロック)と、音楽データファイル(1ブロック)とが示されている。図10では、この2ブロック(16×2=32Kバイト)の各スロットの先頭のバイト(0x0000~0x7FF0)が示されている。図11に分離して示すように、属性ヘッダの先頭から32バイトがヘッダであり、256バイトが曲名領域NM1(256バイト)であり、512バイトが曲名領域NM2(512バイト)である。属性ヘッダのヘッダには、下記のデータが書かれる。

【0052】BLKID-HD0(4バイト)

意味: BLOCKID FILE ID

機能: ATRAC3データファイルの先頭であることを識別するための値

値: 固定値="HD=0"(例えば0x48442D30)

MCode(2バイト)

意味: MAKER CODE

機能: 記録した機器の、メーカー、モデルを識別するコード

値: 上位10ビット(メーカーコード) 下位6ビット(機種コード)

BLOCK SERIAL(4バイト)

意味: トラック毎に付けられた連続番号

機能: ブロックの先頭は0から始まり次のブロックは+1ずつインクリメント編集されても値を変化させない

値: 0より始まり0xFFFFFFFまで。

【0053】N1C+L(2バイト)

意味: トラック(曲名)データ(NM1)の属性を表す

機能: NM1に使用される文字コードと言語コードを各1バイトで表す

値: SN1C+Lと同一

N2C+L(2バイト)

意味: トラック(曲名)データ(NM2)の属性を表す

機能: NM2に使用される文字コードと言語コードを各1バイトで表す

値: SN1C+Lと同一

INFSIZE(2バイト)

意味: トラックに関する付加情報の全てを合計したサイズを表す

機能: データサイズを16バイト単位の大きさで記述、無い場合は必ずオールゼロとすること

値: サイズは0x0000から0x3C6(966)

T-PRT(2バイト)

意味: トータルパーツ数

機能: トラックを構成するパーツ数を表す。通常は1

値: 1から0x285(645dec)

T-SU(4バイト)

意味: トータルSU数

機能: 1トラック中の実際の総SU数を表す。曲の演奏

時間に相当する

値: 0x01から0x001FFFFFFF

INX(2バイト)(Option)

意味: INDEXの相対場所

機能: 曲のさびの部分(特徴的な部分)の先頭を示すポインタ。曲の先頭からの位置をSUの個数を1/4した数で指定する。これは、通常のSUの4倍の長さの時間(約93m秒)に相当する

値: 0から0xFFFF(最大、約6084秒)

XT(2バイト)(Option)

意味: INDEXの再生時間

機能: INX-nnnで指定された先頭から再生すべき時間のSUの個数を1/4した数で指定する。これは、通常のSUの4倍の長さの時間(約93m秒)に相当する

値: 0x0000: 無設定 0x01から0xFFFFE(最大6084秒)

0xFFFF: 曲の終わりまで。

【0054】次に曲名領域NM1およびNM2について説明する。

【0055】NM1

意味: 曲名を表す文字列

機能: 1バイトの文字コードで表した可変長の曲名(最大で256)

名前データの終了は、必ず終端コード(0x00)を書き込むこと

サイズはこの終端コードから計算すること、データの無い場合は少なくとも先頭(0x0020)からヌル(0x00)を1バイト以上記録すること

値: 各種文字コード

NM2

意味: 曲名を表す文字列

機能: 2バイトの文字コードで表した可変長の名前データ(最大で512)

名前データの終了は、必ず終端コード(0x00)を書き込むこと

サイズはこの終端コードから計算すること、データの無い場合は少なくとも先頭(0x0120)からヌル(0x00)を2バイト以上記録すること

値: 各種文字コード。

【0056】属性ヘッダの固定位置(0x320)から始まる、80バイトのデータをトラック情報領域TRKINFと呼び、主としてセキュリティ関係、コピー制御関係の情報を一括して管理する。図12にTRKINFの部分を示す。TRKINF内のデータについて、配置順序に従って以下に説明する。

【0057】CONTENTS KEY(8バイト)

意味: 曲毎に用意された値で、メモ리카ードのセキュリティブロックで保護されてから保存される

機能: 曲を再生する時、まず必要となる最初の鍵となる。MAC計算時に使用される

値：0から0xFFFFFFFFFFFFFFFFFまで
MAC (8バイト)

意味：著作権情報改ざんチェック値

機能：コンテンツ累積番号を含む複数のTRKINFの内容と隠しシーケンス番号から作成される値

隠しシーケンス番号とは、メモ리카ードの隠し領域に記録されているシーケンス番号のことである。著作権対応でないレコーダは、隠し領域を読むことができない。また、著作権対応の専用のレコーダ、またはメモ리카ードを読むことを可能とするアプリケーションを搭載したパーソナルコンピュータは、隠し領域をアクセスすることができる。

【0058】A (1バイト)

意味：パーツの属性

機能：パーツ内の圧縮モード等の情報を示す

値：図13を参照して以下に説明する

ただし、N=0, 1のモノラルは、bit7が1でサブ信号を0、メイン信号(L+R)のみの特別なJointモードをモノラルとして規定する。bit2, 1の情報は通常の再生機は無視しても構わない。

【0059】Aのビット0は、エンファシスのオン/オフの情報を形成し、ビット1は、再生SKIPか、通常再生かの情報を形成し、ビット2は、データ区分、例えばオーディオデータか、FAX等の他のデータかの情報を形成する。ビット3は、未定義である。ビット4、5、6を組み合わせることによって、図示のように、ATRAC3のモード情報が規定される。すなわち、Nは、この3ビットで表されるモードの値であり、モノ(N=0, 1), LP(N=2), SP(N=4), EX(N=5), HQ(N=7)の5種類のモードについて、記録時間(64MBのメモ리카ードの場合)、データ転送レート、1ブロック内のSU数がそれぞれ示されている。1SUのバイト数は、(モノ:136バイト、LP:192バイト、SP:304バイト、EX:384バイト、HQ:512バイト)である。さらに、ビット7によって、ATRAC3のモード(0: Dual 1: Joint)が示される。

【0060】一例として、64MBのメモ리카ードを使用し、SPモードの場合について説明する。64MBのメモ리카ードには、3968ブロックがある。SPモードでは、1SUが304バイトであるので、1ブロックに53SUが存在する。1SUは、(1024/44100)秒に相当する。従って、1ブロックは、 $(1024/44100) \times 53 \times (3968-16) = 4863 \text{ 秒} = 81 \text{ 分}$

転送レートは、

$$(44100/1024) \times 304 \times 8 = 104737$$

bps

となる。

【0061】LT (1バイト)

意味：再生制限フラグ(ビット7およびビット6)とセキュリティバージョン(ビット5～ビット0)

機能：このトラックに関して制限事項があることを表す

値：ビット7: 0=制限なし 1=制限有り

ビット6: 0=期限内 1=期限切れ

ビット5～ビット0: セキュリティバージョン0 (0以外であれば再生禁止とする)

FNo (2バイト)

意味：ファイル番号

機能：最初に記録された時のトラック番号、且つこの値は、メモ리카ード内の隠し領域に記録されたMAC計算用の値の位置を特定する

値：1から0x190 (400)

MG(D) SERIAL-*nnn* (16バイト)

意味：記録機器のセキュリティブロック(セキュリティIC20)のシリアル番号

機能：記録機器ごとに全て異なる固有の値

値：0から0xFFFFFFFFFFFFFFFFFFFFFFFF

CONNUM (4バイト)

意味：コンテンツ累積番号

機能：曲毎に累積されていく固有の値で記録機器のセキュリティブロックによって管理される。2の32乗、42億曲分用意されており、記録した曲の識別に使用する
値：0から0xFFFFFFFF

【0062】YMDhms-S (4バイト) (Option)

意味：再生制限付きのトラックの再生開始日時

機能：EMDで指定する再生開始を許可する日時

値：上述した日時の表記と同じ

YMDhms-E (4バイト) (Option)

意味：再生制限付きのトラックの再生終了日時

機能：EMDで指定する再生許可を終了する日時

値：上述した日時の表記と同じ

MT (1バイト) (Option)

意味：再生許可回数の最大値

機能：EMDで指定される最大の再生回数

値：1から0xFF 未使用の時は、0x00

LTのbit7の値が0の場合はMTの値は00とすること

CT (1バイト) (Option)

意味：再生回数

機能：再生許可された回数の中で、実際に再生できる回数。再生の度にデクリメントする

値：0x00～0xFF 未使用の時は、0x00である

LTのbit7が1でCTの値が00の場合は再生を禁止すること。

【0063】CC (1バイト)

意味：COPY CONTROL

機能：コピー制御

値：図14に示すように、ビット6および7によってコピー制御情報を表し、ビット4および5によって高速デジタルコピーに関するコピー制御情報を表し、ビット2および3によってセキュリティブロック認証レベルを表す。ビット0および1は、未定義

CCの例：(bit 7, 6) 11：無制限のコピーを許可、01：コピー禁止、00：1回のコピーを許可

(bit 3, 2) 00：アナログないしデジタルインからの録音、MG認証レベルは0とする

CDからのデジタル録音では(bit 7, 6)は00、(bit 3, 2)は00となる

CN (1バイト) (Option)

意味：高速デジタルコピーHSCMS (High speed Serial Copy Management System)におけるコピー許可回数

機能：コピー1回か、コピーフリーかの区別を拡張し、回数で指定する。コピー第1世代の場合にのみ有効であり、コピーごとに減算する

値：00：コピー禁止、01から0xFE：回数、0xFF：回数無制限。

【0064】上述したトラック情報領域TRKINFに続いて、0x0370から始まる24バイトのデータをパーツ管理用のパーツ情報領域PRTINFと呼び、1つのトラックを複数のパーツで構成する場合に、時間軸の順番にPRTINFを並べていく。図15にPRTINFの部分を示す。PRTINF内のデータについて、配置順序に従って以下に説明する。

【0065】PRTSIZE (4バイト)

意味：パーツサイズ

機能：パーツの大きさを表す。クラスタ：2バイト (最上位)、開始SU：1バイト (上位)、終了SU：1バイト (最下位)

値：クラスタ：1から0x1F40 (8000)、開始SU：0から0xA0 (160)、終了SU：0から0xA0 (160) (但し、SUの数え方は、0、1、2、と0から開始する)

PRTKEY (8バイト)

意味：パーツを暗号化するための値

機能：初期値=0、編集時は編集の規則に従うこと

値：0から0xFFFFFFFFFFFFFFFF

CONNUM0 (4バイト)

意味：最初に作られたコンテンツ累積番号キー

機能：コンテンツをユニークにするためのIDの役割

値：コンテンツ累積番号初期値キーと同じ値とされる。

【0066】ATRAC3データファイルの属性ヘッダ中には、図10に示すように、付加情報INFが含まれる。この付加情報は、開始位置が固定化されていない点を除いて、再生管理ファイル中の付加情報INF-S (図7および図8B参照) と同一である。1つまたは複数のパーツの最後のバイト部分 (4バイト単位) の次を開始位置として付加情報INFのデータが開始する。

【0067】INF

意味：トラックに関する付加情報データ

機能：ヘッダを伴った可変長の付加情報データ。複数の異なる付加情報が並べられることがある。それぞれにIDとデータサイズが付加されている。個々のヘッダを含む付加情報データは、最小16バイト以上で4バイトの整数倍の単位

値：再生管理ファイル中の付加情報INF-Sと同じである。

【0068】上述した属性ヘッダに対して、ATRAC3データファイルの各ブロックのデータが続く。図16に示すように、ブロック毎にヘッダが付加される。各ブロックのデータについて以下に説明する。

【0069】BLKID-A3D (4バイト)

意味：BLOCKID FILE ID

機能：ATRAC3データの先頭であることを識別するための値

値：固定値="A3D" (例えば0x41334420)

MCODE (2バイト)

意味：MAKER CODE

機能：記録した機器の、メーカー、モデルを識別するコード

値：上位10ビット (メーカーコード) 下位6ビット (機種コード)

CONNUM0 (4バイト)

意味：最初に作られたコンテンツ累積番号

機能：コンテンツをユニークにするためのIDの役割、編集されても値は変化させない

値：コンテンツ累積番号初期値キーと同じ値とされる

BLOCK SERIAL (4バイト)

意味：トラック毎に付けられた連続番号

機能：ブロックの先頭は0から始まり次のブロックは+1ずつインクリメント編集されても値を変化させない

値：0より始まり0xFFFFFFFFFまで

BLOCK-SEED (8バイト)

意味：1ブロックを暗号化するための1つの鍵

機能：ブロックの先頭は、記録機器のセキュリティブロックで乱数を生成、続くブロックは、+1インクリメントされた値、この値が失われると、1ブロックに相当する約1秒間、音が出せないために、ヘッダとブロック末尾に同じものが二重に書かれる。編集されても値を変化させない

値：初期は8バイトの乱数

INITIALIZATION VECTOR (8バイト)

意味：ブロック毎にATRAC3データを暗号化、復号化する時に必要な初期値

機能：ブロックの先頭は0から始まり、次のブロックは最後のSUの最後の暗号化された8バイトの値。デバイ

ドされたブロックの途中からの場合は開始SUの直前の最後の8バイトを用いる。編集されても値を変化させない

値: 0から0xFFFFFFFFFFFFFFFF

SU-nnn

意味: サウンドユニットのデータ

機能: 1024サンプルから圧縮されたデータ、圧縮モードにより出力されるバイト数が異なる。編集されても値を変化させない(一例として、SPモードの時では、N=384バイト)

値: ATRAC3のデータ値。

【0070】図10では、N=384であるので、1ブロックに42SUが書かれる。また、1ブロックの先頭の2つのスロット(4バイト)がヘッダとされ、最後の1スロット(2バイト)にBLKID-A3D、MCODE、CONNUM0、BLOCK SERIALが二重に書かれる。従って、1ブロックの余りの領域Mバイトは、 $(16, 384 - 384 \times 42 - 16 \times 3 = 208)$ (バイト)となる。この中に上述したように、8バイトのBLOCK SEEDが二重に記録される。

【0071】また、サウンドユニットSU(0)~(101)は、図2に示す暗号化/復号ユニット64においてCBC(Cipher Block Chaining)モードで64ビット(8バイト)の暗号化ブロックを単位として暗号化して生成された8バイトの暗号文 C_i によって構成される。本実施形態では、サウンドユニットSUのバイト数(例えば160バイト)を、暗号化の単位である暗号化ブロックのバイト数(例えば8バイト)の整数倍にしている。すなわち、1サウンドユニットSUは例えば20個の暗号文 C_i からなる。このとき、個々の暗号文 C_i は一のサウンドユニットSU内に位置し、一の暗号文 C_i が複数のサウンドユニットSUに跨がって位置することはない。

【0072】ここで、フラッシュメモリ34に記憶されているオーディオデータは、後述するように例えば、ATRAC3方式で圧縮されており、当該圧縮の単位がサウンドユニットSUである。従って、携帯用記憶装置3から携帯用プレーヤ4にオーディオデータを読み出す場合には、読み出しの最小単位は当該サウンドユニットSUとなる。

【0073】このようにすることで、フラッシュメモリ34に記憶されている暗号化されたオーディオデータにアクセスする際に、暗号化ブロックの区切りを意識する必要がなくなり、当該アクセスに伴う処理負担を軽減できる。なお、各クラスタ内に含まれるサウンドユニットSUの数は、1個以上102個以下の範囲で任意である。また、オーディオデータの圧縮方式は、ATRAC3などのATRAC方式以外のCODEC方式でもよい。

$$IK_j = f(MK_j, ID_m)$$

【0074】ブロックシードデータBSは、各ブロック毎に例えば乱数を発生して生成されたデータであり、後述するように、携帯用プレーヤ4内でブロック毎にブロック鍵データBKを生成する際に用いられる。当該ブロックシードデータBSは、エラー対策としてブロック内の2箇所に格納されている。なお、各クラスタに含まれるサウンドユニットは、暗号化された順でフラッシュメモリ34の連続したアドレスに記憶される。また、各サウンドユニット内の暗号化ブロックは、暗号化された順にフラッシュメモリ34の連続したアドレスに記憶される。

【0075】【フラッシュメモリ管理モジュール35】フラッシュメモリ管理モジュール35は、フラッシュメモリ34へのデータの書き込み、フラッシュメモリ34からのデータの読み出しなどの制御を行う。

【0076】携帯用プレーヤ4

図2に示すように、携帯用プレーヤ4は、例えば、主制御モジュール41、通信インターフェイス42、制御モジュール43、編集モジュール44、圧縮/伸長モジュール45、スピーカ46、D/A変換器47およびA/D変換器48を有する。

【0077】【主制御モジュール41】主制御モジュール41は、携帯用プレーヤ4の処理を統括的に制御する。

【0078】【制御モジュール43】図2に示すように、制御モジュール43は、例えば、乱数発生ユニット60、記憶ユニット61、鍵生成/鍵演算ユニット62、相互認証ユニット63、暗号化/復号ユニット64および制御ユニット65を有する。制御モジュール43は、制御モジュール33と同様に、シングルチップの暗号処理専用の集積回路であり、多層構造を有し、内部のメモリセルはアルミニウム層などのダミー層に挟まれている。また、制御モジュール43は、動作電圧または動作周波数の幅が狭く、外部から不正にデータを読み出せないように耐タンパー性を有している。乱数発生ユニット60は、乱数発生指示を受けると、64ビット(8バイト)の乱数を発生する。記憶ユニット61は、認証処理に必要な種々のデータを記憶している。

【0079】図17は、記憶ユニット61に記憶されているデータを説明するための図である。図17に示すように、記憶ユニット61は、マスター鍵データ $MK_0 \sim MK_{31}$ および装置識別データ ID_d を記憶している。ここで、マスター鍵データ $MK_0 \sim MK_{31}$ と、認証鍵データ $IK_0 \sim IK_{31}$ の間には、前述した携帯用記憶装置3の装置識別データ ID_m を用いて、下記式(1)に示す関係がある。なお、下記式において、 $f(a, b)$ は、例えば、引数a, bから値を導出する関数である。

【0080】

【数1】

$$\dots (1)$$

但し、 i は、 $0 \leq j \leq 31$ の整数。

【0081】また、記憶ユニット61における認証鍵データ $IK_0 \sim IK_{31}$ の記憶アドレスは、例えば5ビットで表現され、それぞれ記憶ユニット51におけるマスター鍵データ $MK_0 \sim MK_{31}$ と同じ記憶アドレスが割り当てられている。

【0082】鍵生成/鍵演算ユニット62は、例えば、ISO/IEC 9797のMAC演算方式を用いた演算などの種々の演算を行って鍵データを生成する。このとき、“Block cipher Algorithm”としてFIPS PUB 46-2に規定されるDESが用いられる。

【0083】相互認証ユニット63は、例えば、コンピュータ2から入力したオーディオデータを携帯用記憶装置3に出力する動作を行うのに先立って、携帯用記憶装置3との間で相互認証処理を行う。また、相互認証ユニット63は、携帯用記憶装置3からオーディオデータを入力する動作を行うのに先立って、携帯用記憶装置3との間で相互認証処理を行う。また、相互認証ユニット63は、相互認証処理において、前述したMAC演算を行う。当該相互認証処理では、記憶ユニット61に記憶されているデータが用いられる。なお、相互認証ユニット

$$C_i = E_k (P_i \text{ XOR } C_{i-1})$$

i : 1以上の整数

P_i : 平文 (64ビット)

C_i : 暗号文 (64ビット)

XOR : 排他的論理和

E_k : 56ビットの鍵データ k を用いたDES方式の暗号処理

上記式(2)の演算は、図18で表現される。なお、図18において、「IV」は、ブロック暗号化初期値(64ビット)であり、図2に示す携帯用記憶装置3のフラッシュメモリ34において、図8に示すようにクラスタCL内のサウンドユニットSU(0)の直前に記憶される。

【0086】なお、コンピュータ2あるいはCDプレーヤ7から入力したオーディオデータ(平文)は、ATRAC(Adaptive Transform Audio Coder)方式を改良したATRAC3方式で圧縮されている。なお、ATRACは、MD(Mini Disk: 商標)のための符号化圧縮方式で

$$P_i = C_{i-1} \text{ XOR } D_k (C_i)$$

i : 1以上の整数

P_i : 平文 (64ビット)

C_i : 暗号文 (64ビット)

XOR : 排他的論理和

D_k : 56ビットの鍵データ k を用いたDES方式の復号処理

上記式(3)の演算は、図19で表現される。なお、図19において、「IV」は、ブロック暗号化初期値(64ビット)であり、図2に示す携帯用記憶装置3のフラッシュメモリ34において図8に示すようにクラスタC

63は、必要に応じて、例えば、コンピュータ2あるいはネットワーク5上のコンピュータとの間でオーディオデータの入出力を行う動作に先立って、コンピュータ2あるいはネットワーク5上のコンピュータとの間で相互認証処理を行う。

【0084】暗号化/復号ユニット64は、前述したように、FIPS PUB 81に規定されたECBモードおよびCBCモードを選択的に用いてブロック暗号化を行う。ここで、暗号化/復号ユニット64は、CBCモードにおいて、56ビットの鍵データ k を用いて、コンピュータ2あるいはCDプレーヤ7から入力したオーディオデータ(平文)を、64ビットからなる暗号化ブロックを単位として下記式(2)に基づいて暗号化して暗号化されたオーディオデータ(暗号文)を生成する。下記式(2)から分かるように、CBCモードでは、一つ前の暗号文と次の平文との排他的論理和を暗号化するため、同一の平文が入力されても異なる暗号文が出力され、解読が困難であるという利点がある。

【0085】

【数2】

... (2)

あり、例えば、288 kbit/sで44.1 kHz サンプルのステレオ信号が、帯域分割とMDCT(Modified Discrete Cosine Transform)とを併用して符号化されている。すなわち、まず、帯域分割フィルタで1/4, 1/4, 1/2の3つの帯域に分割され、それぞれの帯域の信号がダウンサンプルされ、時間領域の信号としてMDCTで周波数領域に変換され、当該MDCTの係数が適応ビット配分を行ってスカラ量子化されている。

【0087】暗号化/復号ユニット64は、FIPS 81のモードのうち、前述したECBモードおよびCBCモードの復号を選択的に行う。ここで、暗号化/復号ユニット64は、CBCモードにおいて、56ビットの鍵データ k を用いて、暗号文を、64ビットからなる暗号化ブロックを単位として下記式(3)に基づいて復号して平文を生成する。

【0088】

【数3】

... (3)

L内のサウンドユニットSU(0)の直前に記憶されたものが用いられる。

【0089】制御ユニット65は、乱数発生ユニット60、記憶ユニット61、鍵生成/鍵演算ユニット62、相互認証ユニット63および暗号化/復号ユニット64の処理を統括的に制御する。

【0090】[編集モジュール44] 編集モジュール44は、例えば、図4に示すように携帯用記憶装置3のフラッシュメモリ34内に記憶されたトラックデータファイル1010 ~ 1013を、ユーザからの操作指示に基

づいて編集して新たなトラックデータファイルを生成する。当該編集には、1個のトラックデータファイルを分割して2個のトラックデータファイルを生成する分割編集処理と、2個のトラックデータファイルを結合して1個のトラックデータファイルを生成する結合編集処理とがある。なお、当該編集にあたって、再生管理ファイル100およびトラックデータファイル101₀～101₃が書き換えられる。編集モジュール44における編集処理については後に詳細に説明する。

【0091】〔圧縮／伸長モジュール45〕圧縮／伸長モジュール45は、例えば、携帯用記憶装置3から入力した暗号化されたオーディオデータを復号した後に再生する際に、ATRAC3方式で圧縮されているオーディオデータを伸長し、当該伸長したオーディオデータをD/A変換器47に出力する。また、例えば、CDプレーヤ7あるいはコンピュータ2から入力したオーディオデータを、携帯用記憶装置3に記憶する際に、当該オーディオデータをATRAC3方式で圧縮する。

【0092】〔D/A変換器47〕D/A変換器47は、圧縮／伸長モジュール45から入力したデジタル形式のオーディオデータをアナログ形式のオーディオデータに変換してスピーカ46に出力する。

【0093】〔スピーカ46〕スピーカ46は、D/A変換器47から入力したオーディオデータに応じた音響を出力する。

【0094】〔A/D変換器48〕A/D変換器48は、例えば、CDプレーヤ7から入力したアナログ形式のオーディオデータをデジタル形式に変換して圧縮／伸長モジュール45に出力する。

【0095】以下、図1に示すオーディオシステム1の動作について説明する。

【0096】携帯用記憶装置3への書き込み動作
図20は、携帯用プレーヤ4から携帯用記憶装置3への書き込み動作を説明するためのフローチャートである。

【0097】ステップS1：携帯用プレーヤ4から携帯用記憶装置3に、書き込み要求信号が出力される。

【0098】ステップS2：携帯用記憶装置3と携帯用プレーヤ4との間で、相互認証処理を行う際に用いる認証鍵データIK_jの選択処理が行われる。当該処理については後述する。

【0099】ステップS3：携帯用記憶装置3と携帯用プレーヤ4との間で相互認証処理が行われる。当該処理については後述する。

【0100】ステップS4：ステップS3の相互認証処

$$IK_j = f(MK_j, ID_m)$$

これにより、携帯用記憶装置3と携帯用プレーヤ4とが、上記式(4)に示す関係を持つ認証鍵データIK₀～IK₃₁およびマスター鍵データMK₀～MK₃₁を有している場合には、図21に示す処理によって同じ認証鍵データIK_jが選択される。当該選択された認証鍵データIK_jは、後述する相互認証処理を行う際に、秘密鍵として用いられる。また、このとき、32個の認証鍵データIK_jのうち選択される認証鍵データは、図21に示す処理を行う毎に乱数R_jに応じてランダムに決定される。そのため、不正な認証が成功する確率を、一の認

理によって携帯用記憶装置3および携帯用プレーヤ4の双方が相手を正当であると認めた場合には、ステップS5の処理が行われ、そうでない場合には処理が終了する。

【0101】ステップS5：携帯用記憶装置3および携帯用プレーヤ4において、セッション鍵データSekが生成される。当該処理については後述する。

【0102】ステップS6：携帯用プレーヤ4から携帯用記憶装置3に、通信インターフェイス32、42を介して、暗号化したオーディオデータを出力して書き込む。当該処理については後述する。

【0103】このように、オーディオシステム1によれば、携帯用記憶装置3と携帯用プレーヤ4との間で相互認証が行われ、双方が相手を正当であると認めた場合にのみ、携帯用プレーヤ4から携帯用記憶装置3に、暗号化されたオーディオデータが書き込まれる。そのため、著作権侵害を招くようなオーディオデータの不正な複製が容易に行われることを回避できる。

【0104】〔認証鍵データIK_jの選択処理（図20に示すステップS2）〕図21は、認証鍵データIK_jの選択処理を説明するための図である。図21に示すように、図2に示す携帯用プレーヤ4の乱数発生ユニット60によって64ビットの乱数R_jが生成される。当該乱数R_jは、携帯用プレーヤ4から携帯用記憶装置3に出力される。そして、携帯用記憶装置3の相互認証ユニット53によって、64ビットの乱数R_jの下位5ビットを用いて、記憶ユニット51に記憶されている認証鍵データIK₀～IK₃₁のうちの認証鍵データIK_j（jは0≤j≤31を満たす整数）が特定される。

【0105】また、携帯用記憶装置3の記憶ユニット51から読み出された装置識別データID_mが、携帯用記憶装置3から携帯用プレーヤ4に出力される。

【0106】そして、携帯用プレーヤ4の相互認証ユニット63によって、乱数R_jの下位5ビットを用いて、マスター鍵データMK₀～MK₃₁のうちのマスター鍵データMK_jが特定される。

【0107】そして、鍵生成／鍵演算ユニット62において、前記特定されたマスター鍵データMK_jと、携帯用記憶装置3から入力した装置識別データID_mとを用いて、下記式(4)に基づいて、認証鍵データIK_jを生成する。下記式(4)において、f(a, b)は、例えば、引数a, bから値を導出する任意の関数である。

【0108】

【数4】

$$\dots (4)$$

タIK_jは、後述する相互認証処理を行う際に、秘密鍵として用いられる。また、このとき、32個の認証鍵データIK_jのうち選択される認証鍵データは、図21に示す処理を行う毎に乱数R_jに応じてランダムに決定される。そのため、不正な認証が成功する確率を、一の認

証鍵データを固定して用いる場合の1/32倍にすることができ、不正な認証が行われることを高い確率で回避できる。

【0109】なお、上述した実施形態では、乱数を用いて8個の認証鍵データ IK_j のうちの認証鍵データを選択する場合を例示したが、携帯用記憶装置3および携帯用プレーヤ4の外部から入力した鍵指定信号に基づいて選択する認証鍵データを決定してもよい。

【0110】〔携帯用記憶装置3と携帯用プレーヤ4との間の相互認証処理（図20に示すステップS3）〕図22は、携帯用記憶装置3と携帯用プレーヤ4との間の相互認証処理を説明するための図である。なお、当該相互認証処理を開始するときには、前述した図21に示す認証鍵データ IK_j の選択処理が終了しており、携帯用プレーヤ4の相互認証ユニット53と携帯用記憶装置3の相互認証ユニット63は、選択した認証鍵データ IK_j 、携帯用記憶装置3の装置識別データ ID_m を有して

$$MAC_A = MAC (IK_j, Rd \parallel R_{ms} \parallel ID_m) \quad \dots (5)$$

ステップS13：携帯用プレーヤ4は、「 $Rd \parallel Sd \parallel MAC_A \parallel j$ 」を携帯用記憶装置3に出力する。

【0116】ステップS14：携帯用記憶装置3の相互認証ユニット53において、図20に示すステップS2で得た認証鍵データ IK_j および「 $Rd \parallel R_{ms} \parallel I$

$$D_m$$
」を用いて、下記式（6）に基づいてMAC演算を行い、 MAC_B を求める。

ステップS15：携帯用記憶装置3の相互認証ユニット53において、ステップS14で求めた MAC_B とステップS13で入力した MAC_A とを比較し、一致していれば、携帯用プレーヤ4が適切な認証鍵データ IK_j を有していることが分かるため、携帯用記憶装置3は携帯用プレーヤ4が正当な相手であると認証する。

【0118】ステップS16：携帯用記憶装置3の相互

$$MAC_C = MAC (IK_j, R_{ms} \parallel Rd) \quad \dots (7)$$

ステップS17：携帯用記憶装置3の乱数発生ユニット50において、64ビットの乱数 S_{ms} を生成する。

【0120】ステップ18：携帯用記憶装置3から携帯用プレーヤ4に、「 $S_{ms} \parallel MAC_C$ 」を出力する。

【0121】ステップS19：携帯用プレーヤ4の相互

$$MAC_d = MAC (IK_j, R_{ms} \parallel Rd) \quad \dots (8)$$

ステップS20：携帯用プレーヤ4の相互認証ユニット63において、ステップS19で求めた MAC_d とステップS18で入力した MAC_C とを比較し、一致していれば、携帯用記憶装置3が適切な認証鍵データ IK_j を有していることが分かるため、携帯用プレーヤ4は携帯用記憶装置3が正当な相手であると認証する。以上の処理によって、携帯用記憶装置3と携帯用プレーヤ4との間の相互認証が行われる。

【0123】〔セッション鍵データ Se_k の生成処理（図20に示すステップS5）〕図23は、セッション鍵データ Se_k の生成処理を説明するための図である。なお、当該セッション鍵データ Se_k の生成処理を開始

いる。

【0111】ステップS10：携帯用記憶装置3の乱数発生ユニット50において、64ビットの乱数 R_{ms} を生成し、これを携帯用プレーヤ4に出力する。

【0112】ステップS11：携帯用プレーヤ4の乱数発生ユニット60において、64ビットの乱数 R_d および S_d を生成する。

【0113】ステップS12：携帯用プレーヤ4の相互認証ユニット63において、図20に示すステップS2で得た認証鍵データ IK_j および「 $Rd \parallel R_{ms} \parallel ID_m$ 」を用いて、下記式（5）に基づいてMAC演算を行い、 MAC_A を求める。

【0114】ここで、 $A \parallel B$ は、AとBの連結（nビットのAの後ろにmビットのBを結合して（n+m）ビットとしたもの）を示す。

【0115】

【数5】

D_m 」を用いて、下記式（6）に基づいてMAC演算を行い、 MAC_B を求める。

【0117】

【数6】

認証ユニット53において、図20に示すステップS2で得た認証鍵データ IK_j および「 $R_{ms} \parallel R_d$ 」を用いて、下記式（7）に基づいてMAC演算を行い、 MAC_C を求める。

【0119】

【数7】

認証ユニット63において下記式（8）に基づいてMAC演算を行い、 MAC_d を求める。

【0122】

【数8】

するときには、前述した図21に示す認証鍵データ IK_j の選択処理および図22に示す相互認証処理が終了しており、携帯用記憶装置3および携帯用プレーヤ4の双方は、選択した認証鍵データ IK_j および乱数 S_d 、 S_{ms} を有している。

【0124】ステップS30：携帯用プレーヤ4の相互認証ユニット63は、選択した認証鍵データ IK_j および「 $S_d \parallel S_{ms}$ 」を用いて、下記式（9）に基づいてMAC演算を行い、セッション鍵データ Se_k を生成する。

【0125】

【数9】

$$\text{セッション鍵データ } S_{ek} = \text{MAC} (IK_j, S_d \parallel S_{ms}) \dots (9)$$

ステップS31: 携帯用記憶装置3の相互認証ユニット53は、選択した認証鍵データ IK_j および「 $S_d \parallel S_{ms}$ 」を用いて、下記式(10)に基づいてMAC演算を行い、セッション鍵データ S_{ek} を生成する。当該セッション鍵データ S_{ek} は、正当な相手同士であれば、携

帯用プレーヤ4で生成したセッション鍵データ S_{ek} と同じになる。

【0126】

【数10】

$$\text{セッション鍵データ } S_{ek} = \text{MAC} (IK_j, S_d \parallel S_{ms}) \dots (10)$$

〔携帯用記憶装置3へのオーディオデータの書き込み処理(図20に示すステップS6)〕図24は、携帯用プレーヤ4から携帯用記憶装置3へのオーディオデータの書き込み処理を説明するための図である。なお、当該書き込み処理を開始するときには、前述した図23に示すセッション鍵データ S_{ek} の生成処理は終了しており、携帯用記憶装置3および携帯用プレーヤ4は同じセッション鍵データ S_{ek} を有している。

ら読み出した記憶用鍵データ SK_m を用いて暗号化する。

【0132】ステップS45: 携帯用記憶装置3は、当該暗号化されたコンテンツ鍵データ CK を携帯用プレーヤ4に出力する。

【0133】ステップS46: 携帯用プレーヤ4は、当該暗号化されたコンテンツ鍵データ CK を、トラックデータファイル100n内の $TRKINF$ 内に設定する。

【0134】ステップS47: 携帯用プレーヤ4は、乱数発生ユニット60にパーツ毎に乱数を発生させ、当該乱数に応じたパーツ鍵データ PK を生成する。また、携帯用プレーヤ4は、当該生成したパーツ鍵データ PK を、トラックデータファイル101nの管理データ $PR TINF$ 内に設定する。

【0135】ステップS48: 携帯用プレーヤ4は、例えば、パーツ毎に、鍵生成/演算ユニット62において、下記式(11)に示すように、ステップS47で生成したパーツ鍵データ PK とコンテンツ鍵データ CK との排他的論理和を演算し、当該演算結果をテンポラリ鍵データ TMK とする。なお、テンポラリ鍵データ TMK の生成は、排他的論理和を用いるものには限定されず、例えば、パーツ鍵データ PK とコンテンツ鍵データ CK とを加算する加算演算やその他の関数演算を用いるようにしてもよい。

【0136】

【数11】

$$\dots (11)$$

例えば、鍵生成/鍵演算ユニット62において、下記式(12)に示すように、ステップS46で生成したテンポラリ鍵データ TMK と、ステップS47で生成したブロックシードデータ BS とを用いてMAC演算を行い、ブロック毎にブロック鍵データ BK を生成する。

【0138】

【数12】

$$\dots (12)$$

数については、例えば、“Handbook of Applied Cryptography, CRC Press”などに詳しく記述されている。

【0140】ステップS51: 携帯用プレーヤ4は、コンピュータ2あるいは携帯用プレーヤ4から入力したオーディオデータを、圧縮/伸長モジュール45において、ATRAC3方式で圧縮する。そして、暗号化/復号ユニット64において、ステップS50で生成したブロック鍵データ BK を用いて、前記圧縮したオーディオ

【0127】ステップS40: 携帯用プレーヤ4は、乱数発生ユニット60にトラックデータファイル毎に乱数を発生させ、当該乱数に応じたコンテンツ鍵データ CK を生成する。

【0128】ステップS41: 携帯用プレーヤ4は、暗号化/復号ユニット64において、ステップS40で生成したコンテンツ鍵データ CK を、セッション鍵データ S_{ek} を用いて暗号化する。

【0129】ステップS42: 携帯用プレーヤ4は、ステップS41で暗号化したコンテンツ鍵データ CK を携帯用記憶装置3に出力する。

【0130】ステップS43: 携帯用記憶装置3は、ステップS42で入力した暗号化されたコンテンツ鍵データ CK を、暗号化/復号ユニット54において復号する。

【0131】ステップS44: 携帯用記憶装置3は、暗号化/復号ユニット54において、ステップS43で復号したコンテンツ鍵データ CK を、記憶ユニット51か

$$TMK = PK \text{ XOR } CK$$

ステップS49: 携帯用プレーヤ4は、乱数発生ユニット60にブロック毎に乱数を発生させ、当該乱数に応じたブロックシードデータ BS を生成する。また、携帯用プレーヤ4は、当該生成したブロックシードデータ BS を、当該ブロック内の図10に示す対応する位置に設定する。

【0137】ステップS50: 携帯用プレーヤ4は、例

$$BK = \text{MAC} (TMK, BS)$$

なお、MAC演算の他に、例えば、SHA-1 (Secure Hash Algorithm)、RIPMD-160などの一方向性ハッシュ関数(one-way hash function)の入力に秘密鍵を用いた演算を行ってブロック鍵データ BK を生成してもよい。

【0139】ここで、一方向性関数 f とは、 x より $y = f(x)$ を計算することは容易であるが、逆に y より x を求めることが難しい関数をいう。一方向性ハッシュ関

データをCBCモードで暗号化する。

【0141】ステップS52：携帯用プレーヤ4は、ステップS51で暗号化したオーディオデータに属性ヘッダを付加して、通信インターフェイス32、42を介して、携帯用記憶装置3に出力する。

【0142】ステップS53：携帯用記憶装置3は、ステップS52で入力した暗号化されたオーディオデータと属性ヘッダを、フラッシュメモリ34にそのまま書き込む。以上の処理によって、携帯用プレーヤ4から携帯用プレーヤ4へのオーディオデータの書き込み処理が終了する。なお、ここでは、図4のトラックデータファイル1010～1013についてのみ述べたが、携帯用プレーヤ4は、図4の再生管理ファイルについても同様に適宜更新を行う。

【0143】携帯用記憶装置3からの読み出し動作
図25は、携帯用記憶装置3から携帯用プレーヤ4への読み出し動作を説明するためのフローチャートである。

【0144】ステップS61：携帯用プレーヤ4から携帯用記憶装置3に、読み出しを要求するトラックデータ(曲)を特定した読み出し要求信号が出力される。

【0145】ステップS2：図21を用いて前述したように、携帯用記憶装置3と携帯用プレーヤ4との間で相互認証処理を行う際に用いる認証鍵データIK_jの選択処理が行われる。

【0146】ステップS3：図22を用いて前述したように、携帯用記憶装置3と携帯用プレーヤ4との間で相互認証処理が行われる。

【0147】ステップS4：ステップS3の相互認証処理によって携帯用記憶装置3および携帯用プレーヤ4の双方が相手を正当であると認めた場合には、ステップS5の処理が行われ、そうでない場合には処理が終了する。

【0148】ステップS5：携帯用記憶装置3および携帯用プレーヤ4において、セッション鍵データSekが生成される。

【0149】ステップS63：暗号化されたオーディオデータを、通信インターフェイス32、42を介して、携帯用記憶装置3から携帯用プレーヤ4に読み出す。当該処理については後述する。

【0150】すなわち、オーディオシステム1では、携帯用記憶装置3と携帯用プレーヤ4との間で相互認証が行われ、双方が相手を正当であると認めた場合にのみ、後述するように、携帯用プレーヤ4において、携帯用記憶装置3から携帯用プレーヤ4に出力された暗号化されたコンテンツ鍵データCKを適切なセッション鍵データSekで解読できる。そのため、著作権侵害を招くようなオーディオデータの不正な利用が容易に行われることを回避できる。

【0151】〔携帯用記憶装置3からのオーディオデータの読み出し処理(図25に示すステップS63)〕図

26は、携帯用記憶装置3から携帯用プレーヤ4へのオーディオデータの読み出し処理を説明するための図である。なお、当該読み出し処理は、前述した図20に示す書き込み処理の後に行われるため、図4に示すトラックデータファイル1010～1013には、図10に示すように、TRINFにコンテンツ鍵データCKが設定され、パーツ毎にパーツ鍵データPKが設定され、各クラスタCL内にはブロックシードデータBSが設定されている。また、ステップS5の処理が終了しているため、携帯用記憶装置3および携帯用プレーヤ4は、正当な相手同士であれば、同じセッション鍵データSekを有している。

【0152】ステップS71：携帯用記憶装置3は、フラッシュメモリ34に記憶されている図4に示すトラックデータファイル1010～1013のうち読み出し要求信号で特定されるトラックデータに対応するトラックデータファイルを特定し、当該特定したトラックデータファイルを構成するクラスタ内のオーディオデータを、サウンドユニットSUを単位として読み出して携帯用プレーヤ4に出力する。携帯用記憶装置3は、また、上記トラックデータファイルの属性ヘッダを読み出して携帯用プレーヤ4に出力する。

【0153】ステップS72：携帯用プレーヤ4は、当該入力された属性ヘッダのうち、TRINFから暗号化されたコンテンツ鍵CKを抽出し、携帯用記憶装置3に出力する。

【0154】ステップS73：携帯用記憶装置3の暗号化/復号ユニット54は、ステップS72で入力されたコンテンツ鍵データCKを、記憶ユニット51に記憶されている記憶用鍵データSK_mを用いて復号する。

【0155】ステップS74：携帯用記憶装置3の暗号化/復号ユニット54は、ステップS73で復号したコンテンツ鍵データCKを、図25に示すステップS5で得られたセッション鍵データSekを用いて暗号化する。

【0156】ステップS75：携帯用記憶装置3は、ステップS74で暗号化したコンテンツ鍵データCKを携帯用プレーヤ4に出力する。

【0157】ステップS76：携帯用プレーヤ4の暗号化/復号ユニット64は、ステップS73で携帯用記憶装置3から入力したコンテンツ鍵データCKを、セッション鍵データSekを用いて復号する。

【0158】ステップS77：携帯用プレーヤ4の鍵生成/演算ユニット62は、ステップS76で復号されたコンテンツ鍵データCKと、ステップS71で入力された属性ヘッダの中のPRTINFに含まれるパーツ鍵データPKとの排他的論理和を演算し、当該演算結果をテンポラリ鍵データTMKとする。

【0159】

【数13】

$$TMK = PK \text{ XOR } CK$$

ステップS78: 携帯用プレーヤ4の鍵生成/鍵演算ユニット62において、ステップS76で生成したテンポラリ鍵データTMKと、ステップS71で入力されたトラックデータファイルのクラスタ内の図10に示すブロックシードデータBSとを用いて、下記式(14)に示

$$BK = MAC(TMK, BS)$$

ステップS79: 携帯用プレーヤ4は、暗号化/復号ユニット64において、ステップS78で生成したブロック鍵データBKを用いて、ステップS71で入力したオーディオデータを復号する。このとき、オーディオデータの復号は、各ブロック毎に、それぞれ個別に求められたブロック鍵データBKを用いて行われる。また、復号は、暗号化の単位である8バイトのブロックを単位として行われる。

【0161】ステップS80: 携帯用プレーヤ4は、圧縮/伸長モジュール45において、ステップS79で復号したオーディオデータをATRAC3方式で伸長し、当該伸長したオーディオデータを、D/A変換器47でデジタル形式に変換した後に、スピーカ46に出力する。このとき、圧縮/伸長モジュール45は、ステップS78で復号したオーディオデータを、サウンドユニットSUを単位として伸長する。以上の処理によって、携帯用記憶装置3から携帯用プレーヤ44へのオーディオデータの読み出しおよび再生が終了する。

【0162】〔トラックデータファイルの分割編集処理〕前述したように、携帯用プレーヤ4の編集モジュール44は、1個のトラックデータファイルを分割して2個のトラックデータファイルを生成する分割編集処理と、2個のトラックデータファイルを結合して1個のトラックデータファイルを生成する結合編集処理とを行う。

【0163】先ず、分割編集処理について説明する。図27は、携帯用プレーヤ4の編集モジュール44によるトラックデータファイルの分割編集処理を説明するための図である。編集モジュール44は、例えば、図27Aに示す1個のトラックデータファイル(1)を、図27Bに示すトラックデータファイル(1)と、図27Cに示すトラックデータファイル(2)とに分割する。このとき、分割の区切りとなる最小単位はサウンドユニットSUであり、当該例では、図27Bに示すように、トラックデータファイル(1)のクラスタCL(2)のサウンドユニットSU(3)とSU(4)との間で分割されている。

【0164】当該分割により、分割後のトラックデータファイル(1)のクラスタCL(2)は図28Aに示すようになり、新たに生成されたトラックデータファイル(2)のクラスタCL(0)は図28Bに示すようになる。このとき、図28Bに示すように、トラックデータファイル(2)のクラスタCL(0)のサウンドユニッ

... (13)

すMAC演算を行い、当該演算結果をブロック鍵データBKとする。ブロック鍵データBKは、ブロック毎に求められる。

【0160】

【数14】

... (14)

トSU(0)は分割前のトラックデータファイル(1)のクラスタ(2)のサウンドユニットSU(4)となり、トラックデータファイル(2)のクラスタCL(0)のサウンドユニットSU(1)は分割前のトラックデータファイル(1)のクラスタ(2)のサウンドユニットSU(5)となる。また、図28Bに示すトラックデータファイル(2)のクラスタCL(0)のブロック暗号化初期値IVには、図27A、Bに示すトラックデータファイル(1)のクラスタCL(2)内のサウンドユニットSU(3)の最後の8バイトが設定される。

【0165】本実施形態では、前述したように各クラスタ内において、最初のサウンドユニットSU(0)の直前にブロック暗号化初期値IVを配置したことで、分割の際に、分割位置の直前の8バイトをそのままブロック暗号化初期値IVとして用いれば良く、新たなトラックデータファイルを作成する際の処理を簡単にできる。また、再生時に、サウンドユニットSU(0)と共に、その直前のブロック暗号化初期値IVを読み出せばよいため、再生処理も簡単になる。

【0166】本実施形態では、分割前のトラックデータファイル(1)のコンテンツ鍵データ、パーツ鍵データおよびブロック鍵データは、それぞれCK₁、PK₁およびBK₁である。また、分割後のトラックデータファイル(1)のコンテンツ鍵データ、パーツ鍵データおよびブロック鍵データは、それぞれCK₁'、PK₁'およびBK₁である。また、トラックデータファイル(2)のコンテンツ鍵データ、パーツ鍵データおよびブロック鍵データは、それぞれCK₂、PK₂およびBK₁である。

【0167】図29は、携帯用プレーヤ4の編集モジュール44において、新たなトラックデータファイル

(2)のコンテンツ鍵データおよびパーツ鍵データを生成する方法を説明するための図である。分割により生成された新たなトラックデータファイル(2)は、トラックデータファイル(1)とは別に新たなコンテンツ鍵データCK₂を有する。本実施形態では、パーツ鍵データPK₂を以下に示すように算出することで、ブロック鍵データBK₁を分割前と同じにする。

【0168】ステップS90: 編集モジュール44は、トラックデータファイルの分割指示を入力したか否かを判断し、入力したと判断した場合にはステップS91の処理を実行し、入力していないと判断した場合にはステップS90の処理を繰り返す。

【0169】ステップS91：編集モジュール44は、乱数発生ユニット60に乱数を発生させ、当該乱数に応じたコンテンツ鍵データCK__2を新たに生成する。

【0170】ステップS92：携帯用記憶装置3の暗号化／復号ユニット54において、ステップS91で生成したコンテンツ鍵データCK__2を、記憶ユニット51に記憶されている記憶用鍵データSKmを用いて暗号化する。

$$PK_2 = CK_1 \text{ XOR } PK_1 \text{ XOR } CK_2$$

... (15)

これにより、トラックデータファイル(2)について、前記式(11)に基づいてされるテンポラリ鍵データは、トラックデータファイル(1)のテンポラリ鍵データと同じになり、前記式(12)に基づいて生成されるブロック鍵データも分割前のブロック鍵データBK__1と同じにできる。そのため、トラックデータファイル(2)内のサウンドユニットSUを新たなブロック鍵データを用いて再度暗号化する必要がない。

【0174】ステップS95：編集モジュール44は、ステップS94で生成したパーツ鍵データPK__2を、当該トラックデータファイルPRTINFにそのまま書き込む。

【0175】このように、オーディオシステム1では、分割して新たに生成したトラックデータファイル(2)のコンテンツ鍵データとして、新たなコンテンツ鍵データCK__2を用いた場合でも、上記式(15)に基づいてパーツ鍵データPK__2を生成することで、テンポラリ鍵データを分割前のテンポラリ鍵データと同じにできる。その結果、ブロック鍵データも分割前のブロック鍵データBK__1と同じにでき、トラックデータファイル(2)内のサウンドユニットSUを新たなブロック鍵データを用いて再度暗号化する必要がない。また、同様に、分割後のトラックデータファイル(1)のパーツ鍵データPK__1'も、ブロック鍵データBK__1を変えないように、コンテンツ鍵データCK__1'に応じた決定される。その結果、分割後のトラックデータファイル(1)内のサウンドユニットSUを新たなブロック鍵データを用いて再度暗号化する必要もない。そのため、トラックデータファイルの分割編集に伴い演算量が大幅に増加することを回避できる。なお、ここでは、図4のトラックデータファイルについてのみ述べたが、編集モジュール44は、図4の再生管理ファイル100についても同様に適宜更新を行う。

【0176】次に、トラックデータファイルの結合編集処理について説明する。図30は、携帯用プレーヤ4の編集モジュール44によるトラックデータファイルの結合編集処理を説明するための図である。図30に示すように、編集モジュール44は、例えば、図30Aに示すトラックデータファイル(1)と、図30Bに示すトラックデータファイル(2)とを結合して、図30Cに示

すトラックデータファイル(3)を生成する。

【0171】ステップS93：編集モジュール44は、当該暗号化されたコンテンツ鍵データCK__2を、当該トラックデータファイルのTRKINFに書き込む。

【0172】ステップS94：編集モジュール44は、トラックデータファイル(2)のパーツ鍵データPK__2を下記式(15)に基づいて生成する。

【0173】

【数15】

すトラックデータファイル(3)を生成する。

【0177】当該結合により、結合前のトラックデータファイル(1)からなるパーツ(1)と、結合前のトラックデータファイル(2)からなるパーツ(2)とを含む新たなトラックデータファイル(3)が生成される。また、トラックデータファイル(3)のコンテンツ鍵データとして新たなコンテンツ鍵データCK__3が生成され、パーツ(1)のパーツ鍵データPK__3__1およびパーツ(2)のパーツ鍵データPK__3__2が後述するようにして新たに生成される。また、当該トラックデータファイル(3)のTRKINFおよびPRTINFに、新たに生成された鍵データが後述するように設定される。

【0178】また、パーツ(1)の図6に示すPRTS IZEが示す開始クラスタおよび終了クラスタとして、結合前のトラックデータファイル(1)のクラスタCL(0)およびCL(4)がそれぞれ設定される。また、パーツ(2)のPRTS IZEが示す開始クラスタおよび終了クラスタとして、結合前のトラックデータファイル(2)のクラスタCL(0)およびCL(5)がそれぞれ設定される。

【0179】図31は、携帯用プレーヤ4の編集モジュール44において、新たに生成したトラックデータファイル(3)のパーツ(1)および(2)のパーツ鍵データを生成する処理を説明するための図である。なお、本実施形態では、結合の対象となるトラックデータファイル(1)がコンテンツ鍵データCK__1、パーツ鍵データPK__1およびブロック鍵データBK__1を用いており、トラックデータファイル(2)がコンテンツ鍵データCK__2、パーツ鍵データPK__2およびブロック鍵データBK__2を用いている場合を例示して説明する。

【0180】ここで、トラックデータファイル(3)は新たなコンテンツ鍵データCK__3を得るが、パーツ(1)および(2)のパーツ鍵データを以下に示すように算出することで、各ブロックのブロック鍵データBK__1およびBK__2を結合前と同じにできる。

【0181】ステップS100：編集モジュール44は、トラックデータファイルの結合指示を入力したか否かを判断し、入力したと判断した場合にはステップS101の処理を実行し、入力していないと判断した場合に

はステップS100の処理を繰り返す。

【0182】ステップS101：編集モジュール44は、乱数発生ユニット60に乱数を発生させ、当該乱数に応じたコンテンツ鍵データCK₃を新たに生成する。

【0183】ステップS102：携帯用記憶装置3の暗号化/復号ユニット54において、ステップS101で生成したコンテンツ鍵データCK₃を、記憶ユニット51に記憶されている記憶用鍵データSK_mを用いて暗号化する。

$$PK_3_1 = CK_1 \text{ XOR } PK_1 \text{ XOR } CK_3 \dots (16)$$

これにより、前記式(11)に基づいて生成されるパーツ(1)のテンポラリ鍵データを結合前のトラックデータファイル(1)のテンポラリ鍵データと同じにでき、その結果、前記式(12)に基づいて生成されるパーツ(1)のブロック鍵データも結合前のトラックデータファイル(1)のブロック鍵データBK₁と同じにできる。そのため、パーツ(1)のサウンドユニットSUを新たなブロック鍵データを用いて再度暗号化する必要が

$$PK_3_2 = CK_2 \text{ XOR } PK_2 \text{ XOR } CK_3 \dots (17)$$

これにより、前記式(11)に基づいて生成されるパーツ(2)のテンポラリ鍵データを結合前のトラックデータファイル(2)のテンポラリ鍵データと同じにでき、その結果、前記式(12)に基づいて生成されるパーツ(2)のブロック鍵データも結合前のトラックデータファイル(2)のブロック鍵データBK₂と同じにできる。そのため、パーツ(2)のサウンドユニットSUを新たなブロック鍵データを用いて再度暗号化する必要がない。

【0189】ステップS106：編集モジュール44は、ステップS104で生成したパーツ鍵データPK_{3_1}をトラックデータファイル(3)のパーツ(1)のPRTINFにそのまま書き込む。

【0190】ステップS107：編集モジュール44は、ステップS105で生成したパーツ鍵データPK_{3_2}をトラックデータファイル(3)のパーツ(2)のPRTINFにそのまま書き込む。

【0191】このように、オーディオシステム1では、結合して新たに生成したトラックデータファイル(3)のコンテンツ鍵データとして、新たなコンテンツ鍵データCK₃を用いた場合でも、上記式(16)および(17)に基づいてパーツ鍵データPK_{3_1}およびPK_{3_2}を生成することで、各パーツのテンポラリ鍵データを結合前と同じにできる。その結果、各パーツのブロック鍵データも結合前のブロック鍵データBK₁およびBK₂とそれぞれ同じにでき、パーツ(1)および(2)内のサウンドユニットSUを新たなブロック鍵データを用いて再度暗号化する必要がない。そのた

【0184】ステップS103：編集モジュール44は、当該暗号化されたコンテンツ鍵データCK₃を当該トラックデータファイルのTRKINFに書き込む。

【0185】ステップS104：編集モジュール44は、トラックデータファイル(3)のパーツ(1)のパーツ鍵データPK_{3_1}を下記式(16)に基づいて生成する。

【0186】

【数16】

ない。

【0187】ステップS105：編集モジュール44は、トラックデータファイル(3)のパーツ(2)のパーツ鍵データPK_{3_2}を下記式(17)に基づいて生成する。

【0188】

【数17】

め、トラックデータファイルの結合編集に伴い演算量が大幅に増加することを回避できる。なお、ここでは、図4のトラックデータファイルについてのみ述べたが、編集モジュール44は、図4の再生管理ファイルについても同様に適宜更新を行う。

【0192】この発明は、上述した実施形態等に限定されるものではなく、この発明の要旨を逸脱しない範囲内で様々な変形や応用が可能である。例えば、上述した実施形態では、ATRAC3方式の圧縮の単位であるサウンドユニットSUのバイト数(160バイト)が、CBCモードの暗号化の単位である暗号化ブロックのバイト数(8バイト)の整数倍になる場合を例示したが、この発明は、例えば、整数倍にならない場合には、サウンドユニットSUにデータ長調整用のデータであるパディング(padding)を挿入して調整するようにしてもよい。

【0193】また、上述した実施形態では、携帯用記憶装置3と携帯用プレーヤ4との間で相互認証処理を行う場合に、図22に示すように、先ず始めに携帯用記憶装置3で生成した乱数R_{ms}を携帯用プレーヤ4に出力する場合を例示したが、先ず始めに携帯用プレーヤ4で生成した乱数を携帯用記憶装置3に出力するようにしてもよい。

【0194】また、上述した実施形態では、図21に示すように、記憶ユニット51および61に32組の認証鍵データおよびマスター鍵データを記憶した場合を例示したが、これらの組の数は2以上であれば任意である。

【0195】また、上述した実施形態では、図21に示すように、携帯用プレーヤ4において、マスター鍵デー

タMK₀～MK₃₁から認証鍵データIK₀～IK₃₁を生成する場合を例示したが、携帯用プレーヤ4に、携帯用記憶装置3と同じように、認証鍵データIK₀～IK₃₁を記憶し、乱数R_jに応じた認証鍵データを選択するようにしてもよい。

【0196】また、上述した実施形態では、図21に示すように、携帯用記憶装置3および携帯用プレーヤ4において、携帯用プレーヤ4で生成した乱数R_jを用いて、認証鍵データIK_jおよびマスター鍵データMK_jを選択する場合を例示したが、携帯用記憶装置3で生成した乱数を用いてもよいし、携帯用記憶装置3および携帯用プレーヤ4の双方で発生した乱数を用いてもよい。

【0197】また、上述した実施形態では、図21に示すように、携帯用記憶装置3および携帯用プレーヤ4において乱数R_jに基づいて認証鍵データIK_jおよびマスター鍵データMK_jを選択する場合を例示したが、この発明は、例えば、携帯用記憶装置3および携帯用プレーヤ4に外部から5ビットの鍵選択指示データを入力し、当該鍵選択指示データで指示される相互に対応する認証鍵データIK_jおよびマスター鍵データMK_jを、携帯用記憶装置3および携帯用プレーヤ4で選択してもよい。

【0198】また、上述した実施形態では、トラックデータとしてオーディオデータを含むデータを例示したが、この発明は、その他、動画像データ、静止画像データ、文書データおよびプログラムデータなどを含むトラックデータをフラッシュメモリ34に記憶する場合にも適用できる。

【0199】

【発明の効果】以上説明したように、この発明のデータ処理システムおよびその方法によれば、トラックデータを第3の鍵データを用いて暗号化し記憶装置に記憶した後に、第1の鍵データが変更された場合でも、第3の鍵データが変更されることはなく、トラックデータを復号して再度暗号化を行う必要がなくなる。そのため、第1の鍵データが変更された場合のデータ処理量を大幅に削減できる。

【図面の簡単な説明】

【図1】この発明の一実施形態のオーディオシステムのシステム構成を示すブロック図である。

【図2】携帯用記憶装置および携帯用プレーヤの内部構成を示すブロック図である。

【図3】携帯用記憶装置内の記憶ユニットに記憶されているデータを説明するための略線図である。

【図4】携帯用記憶装置のフラッシュメモリに記憶されているデータを説明するための略線図である。

【図5】再生管理ファイルのデータ構成を概略的に示す略線図である。

【図6】データファイルのデータ構成を概略的に示す略線図である。

【図7】再生管理ファイルのデータ構成をより詳細に示す略線図である。

【図8】再生管理ファイルの各部分と付加情報領域の構成を示す略線図である。

【図9】携帯用プレーヤの記憶ユニットに記憶されているデータを説明するための略線図である。

【図10】データファイルのデータ構成をより詳細に示す略線図である。

【図11】データファイルの属性ヘッダの一部を示す略線図である。

【図12】データファイルの属性ヘッダの一部を示す略線図である。

【図13】録音モードの種類と、各録音モードにおける録音時間等を示す略線図である。

【図14】コピー制御情報を説明するための略線図である。

【図15】データファイルの属性ヘッダの一部を示す略線図である。

【図16】データファイルの各データブロックのヘッダを示す略線図である。

【図17】携帯用プレーヤの記憶ユニットに記憶されているデータを説明するための略線図である。

【図18】携帯用プレーヤの暗号化／復号ユニットのCBCモードにおける暗号化処理を説明するための略線図である。

【図19】携帯用プレーヤの暗号化／復号ユニットのCBCモードにおける復号処理を説明するための略線図である。

【図20】携帯用プレーヤから携帯用記憶装置への書き込み動作を説明するためのフローチャートである。

【図21】相互認証ユニットによる認証鍵データIK_jの選択処理を説明するための略線図である。

【図22】携帯用記憶装置と携帯用プレーヤとの間の相互認証処理を説明するためのフローチャートである。

【図23】セッション鍵データS_{ek}の生成処理を説明するための略線図である。

【図24】携帯用プレーヤから携帯用記憶装置へのオーディオデータの書き込み処理を説明するためのフローチャートである。

【図25】携帯用記憶装置から携帯用プレーヤへの読み出し動作を説明するためのフローチャートである。

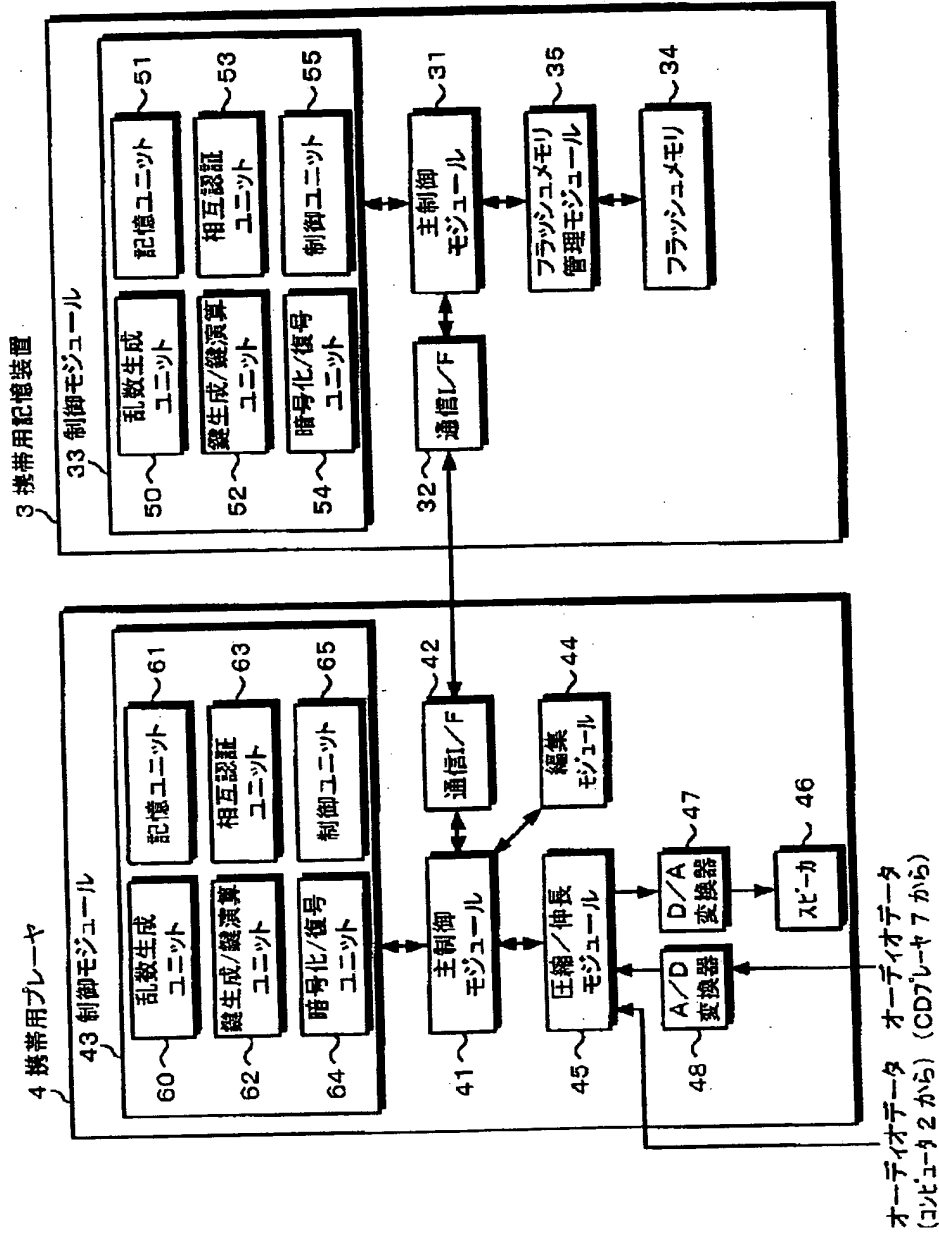
【図26】携帯用記憶装置から携帯用プレーヤへのオーディオデータの読み出し処理を説明するためのフローチャートである。

【図27】携帯用プレーヤの編集モジュールによるトラックデータファイルの分割編集処理を説明するための略線図である。

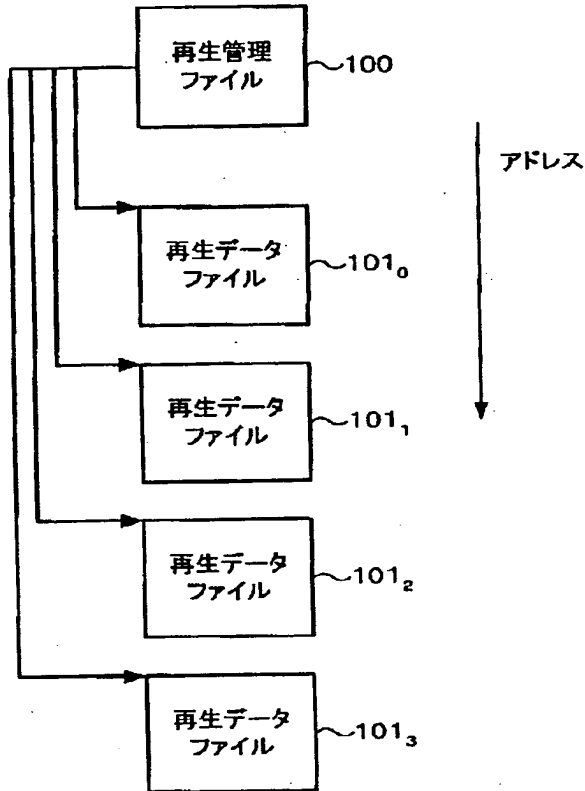
【図28】分割編集処理を行った後のクラスタ内のデータを説明するための略線図である。

【図29】携帯用プレーヤの編集モジュールにおいて、

【図2】



【図4】



【図17】

携帯用プレーヤ4の記録モジュール41に記憶されるデータ

マスター鍵データ MK₀
 MK₁
 MK₂
 MK₃
 ⋮
 MK₃₀
 MK₃₁
 装置識別データ ID₀

携帯用記憶装置3のフラッシュメモリ34の記憶データ

【図7】

再生管理ファイル(PBLIST)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0x0000	BLKID-TLO				Reserved		MCode		REVISION				Reserved			
0x0010	SN1C+L		SN2C+L		SINF SIZE		T-TRK		VerNo		Reserved					
0x0020	NM1-S(256)															
0x0120	NM2-S(512)															
0x0320	Reserved								CONTENTSKEY							
0x0330	Reserved								MAC							
0x0350	Reserved												S-YMDhms			
	TRK-001	TRK-002	TRK-003	TRK-004	TRK-005	TRK-006	TRK-007	TRK-008	TRK-009	TRK-010	TRK-011	TRK-012	TRK-013	TRK-014	TRK-015	TRK-016
	TRK-009	TRK-010	TRK-011	TRK-012	TRK-013	TRK-014	TRK-015	TRK-016								
0x0660	TRK-393	TRK-394	TRK-395	TRK-396	TRK-397	TRK-398	TRK-399	TRK-400								
0x0647	INF-S(14720)															
0x3FF0	BLKID-TLO				Reserved		MCode		REVISION				Reserved			

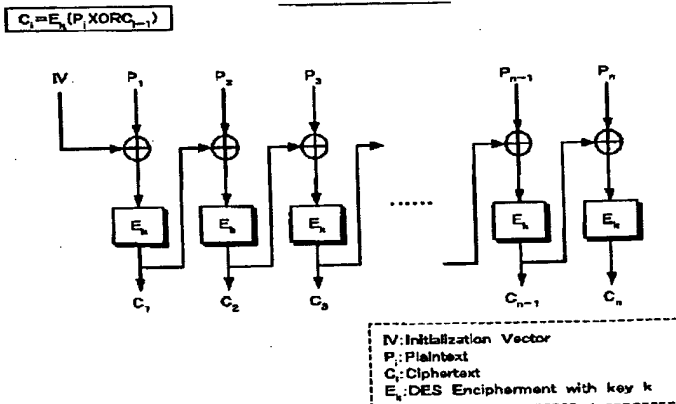
【図10】

A3Dmms.MSA(ATRAC3データファイル)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
0x0000	BLKID-HD0			Reserved		MCCode		Reserved			BLOCK SERIAL							
0x0010	NTCH-L		N2CH-L		INFSIZE		T-PRT		T-SU			INX		XT				
0x0020	NM1(256)																	
0x0120	NM2(512)																	
0x0310	Reserved(8)																	
0x0320	Reserved(8)								CONTENTSKEY									
	Reserved(8)								MAG									
	Reserved(12)												A		LT		FNo	
	MG(D)SERIAL-nnn																	
0x0360	CONNUM			YMDhms-S			YMDhms-E			MT		GT		CC		CN		
0x0370	PRTSIZE			PRTKEY										Reserved(8)				
0x0380				CONNUM0			PRTSIZE(0x0388)			PRTKEY								
0x0390				Reserved(8)										CONNUM0				
	INF(0x0400)																	
0x3FFF	BLKID-HD0			Reserved		MCCode		Reserved			BLOCK SERIAL							
0x4000	BLKID-A3D			Reserved		MCCode		CONNUM0			BLOCK SERIAL							
0x4010	BLOCK SEED								INITIALIZATION VECTOR									
0x4020	SU-000(Nbyte=384byte)																	
0x41A0	SU-001(Nbyte)																	
0x4320	SU-002(Nbyte)																	
0x04A0	SU-041(Nbyte)																	
0x7DA0	Reserved(Nbyte=208byte)																	
0x7F80	Reserved(Nbyte=208byte)																	
	BLOCK SEED																	
0x7FF0	BLKID-A3D			Reserved		MCCode		CONNUM0			BLOCK SERIAL							

【図18】

DES CBCモード(暗号化)



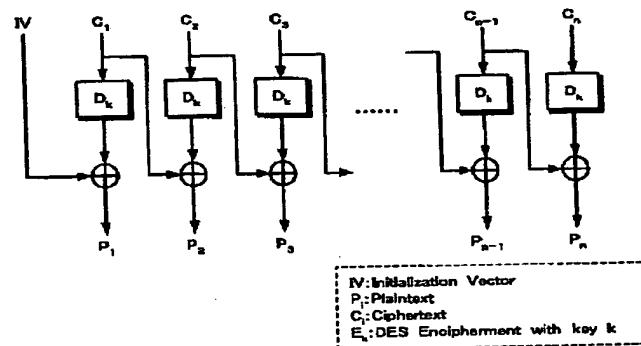
【図16】

0x4000	BLKID-A3D	Reserved	MCCode	CONNUM0	BLOCK SERIAL
0x4010	BLOCK SEED				INITIALIZATION VECTOR
0x4020	SU-000(Nbyte=384byte)				

【図19】

DES CBCモード(復号化)

$$P_i = C_{i-1} \oplus D_k(C_i)$$



【図28】

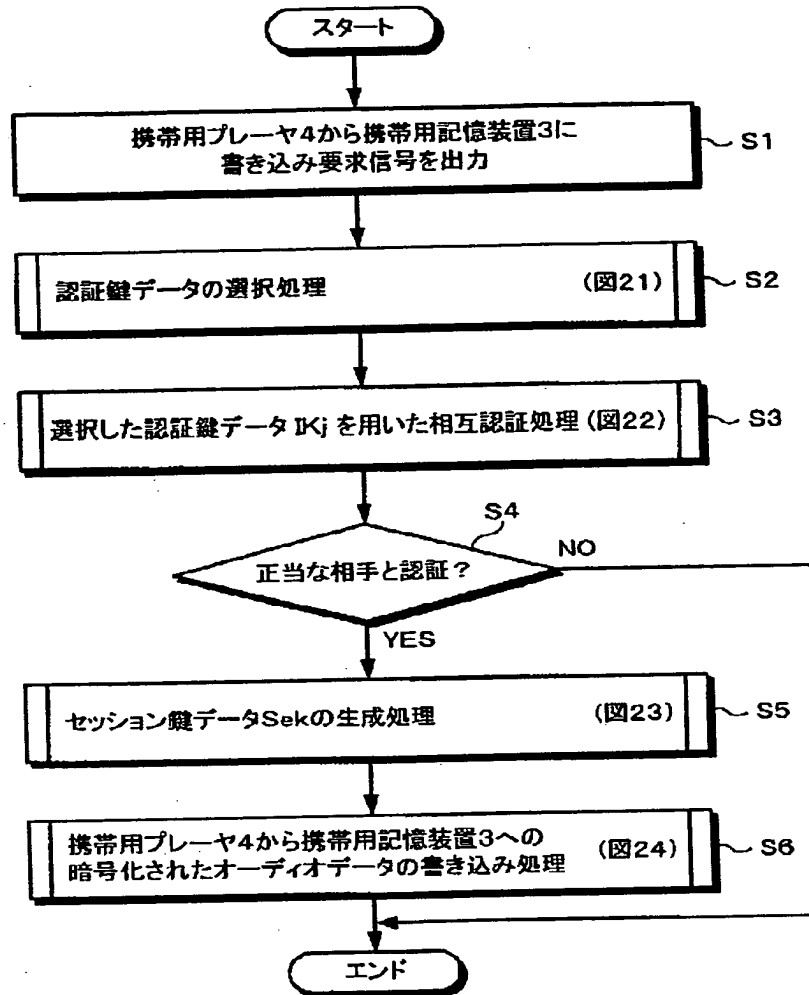
分割後のトラック(1)のクラスタCL(2)

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
BLKID-A3D				Reserved		MCode		CONNUM0				BLOCK SERIAL			
BLOCK SEED								INITIALIZATION VECTOR (DL(1)の最後のSUの最後のバイト)							
SU (0)															
SU (1)															
SU (2)															
SU (3)															
BLOCK SEED															
BLKID-A3D				Reserved		MCode		CONNUM0				BLOCK SERIAL			

トラック(2)のクラスタCL(0)

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
BLKID-A3D				Reserved		MCode		CONNUM0				BLOCK SERIAL			
BLOCK SEED								INITIALIZATION VECTOR (トラック(1)のDL(2)のSU(3)の最後のバイト)							
SU (0) (分割前のトラック(1)のDL(2)のSU(4))															
SU (1) (分割前のトラック(1)のDL(2)のSU(5))															
BLOCK SEED															
BLKID-A3D				Reserved		MCode		CONNUM0				BLOCK SERIAL			

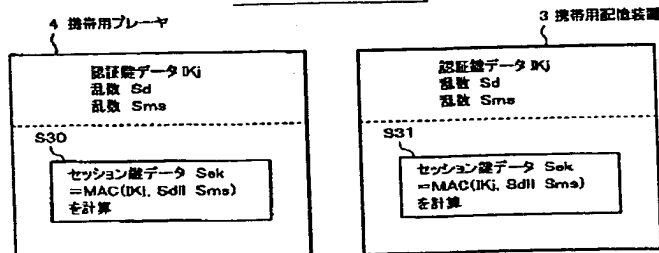
【図20】



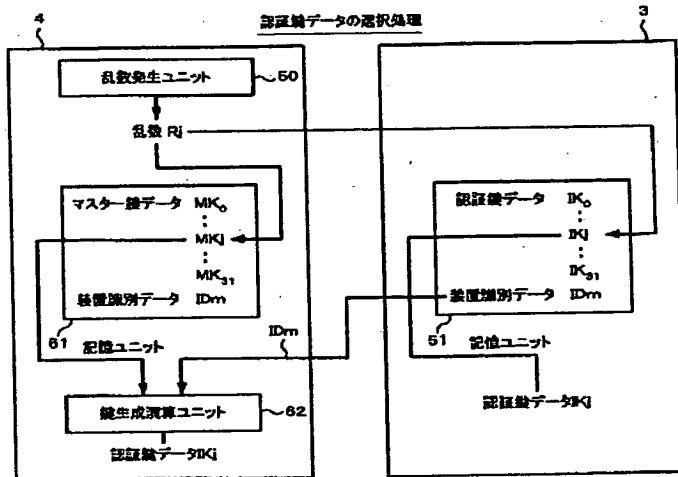
携帯用記憶装置3への書込処理

【図23】

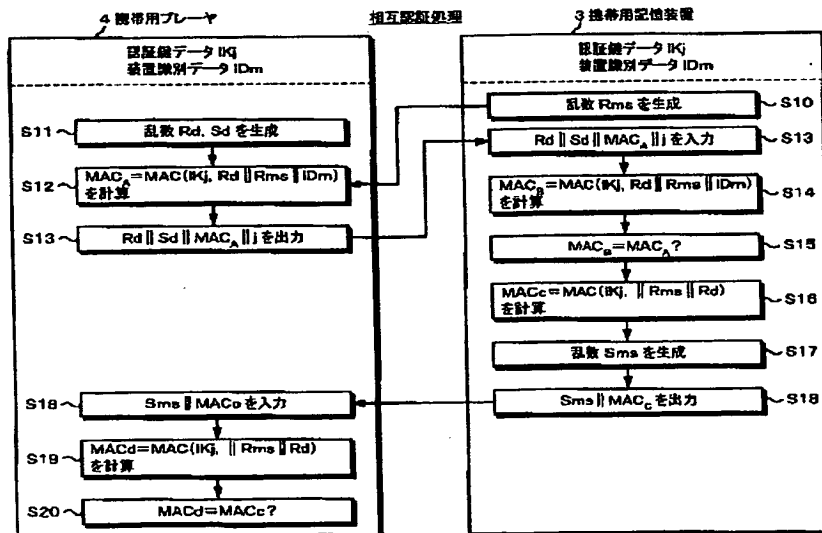
セッション鍵データの生成処理



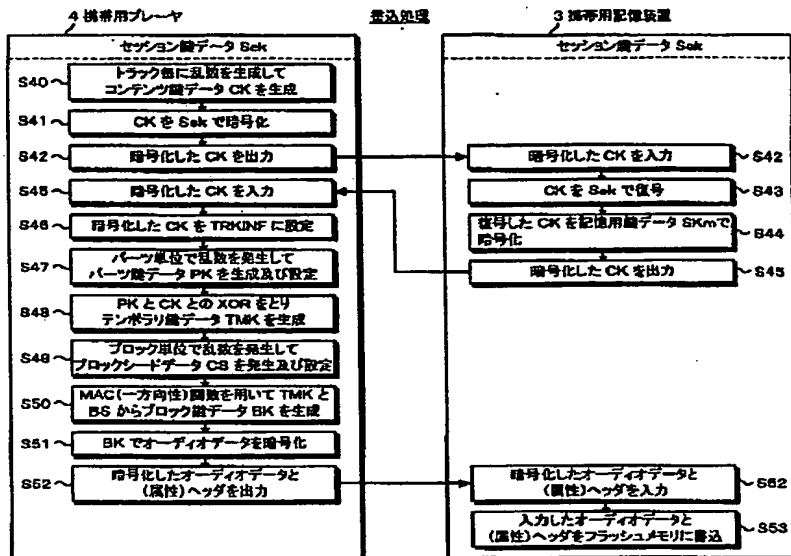
【図21】



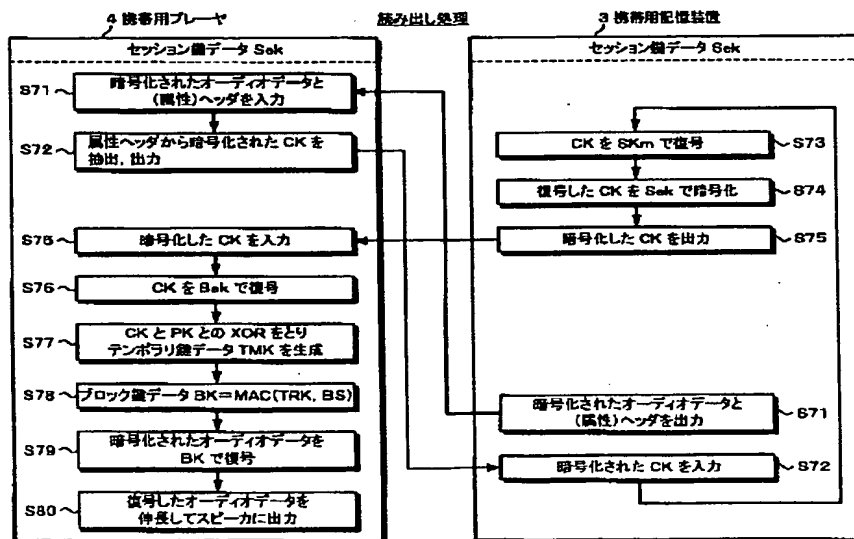
【図22】



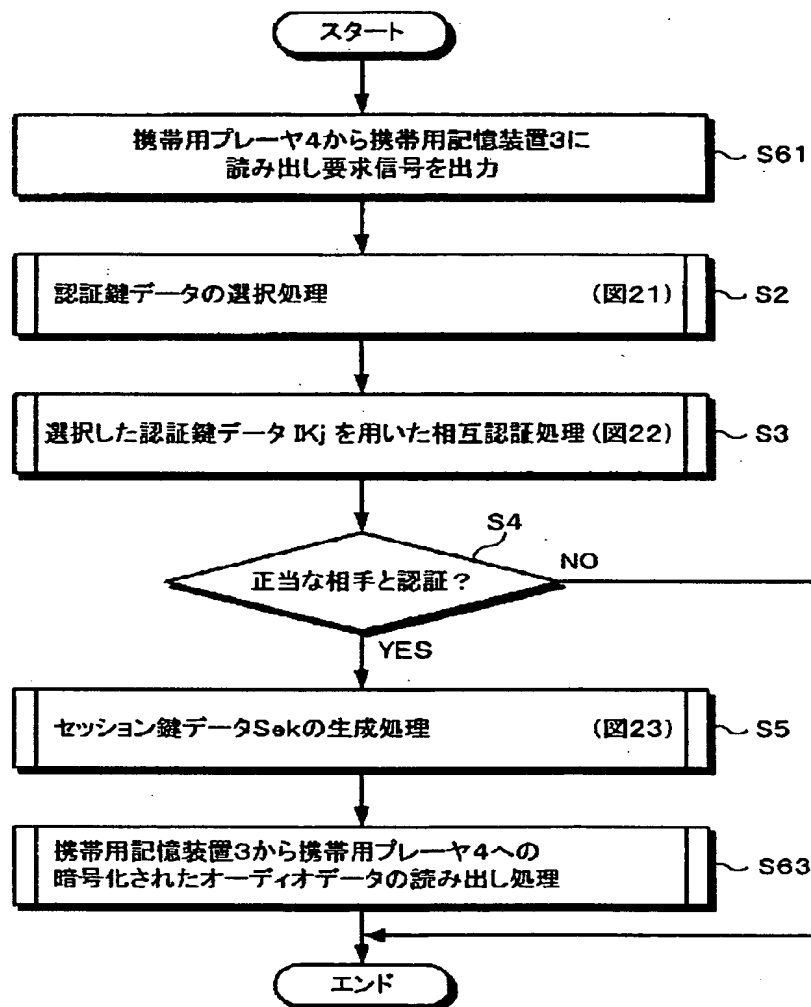
【図24】



【図26】

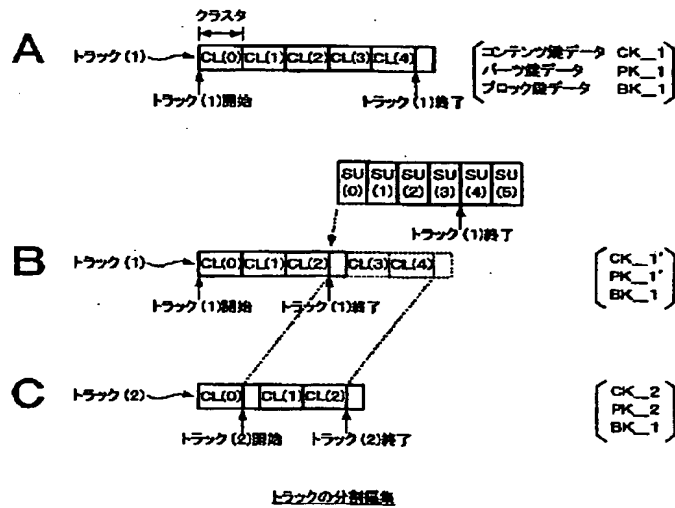


【図25】

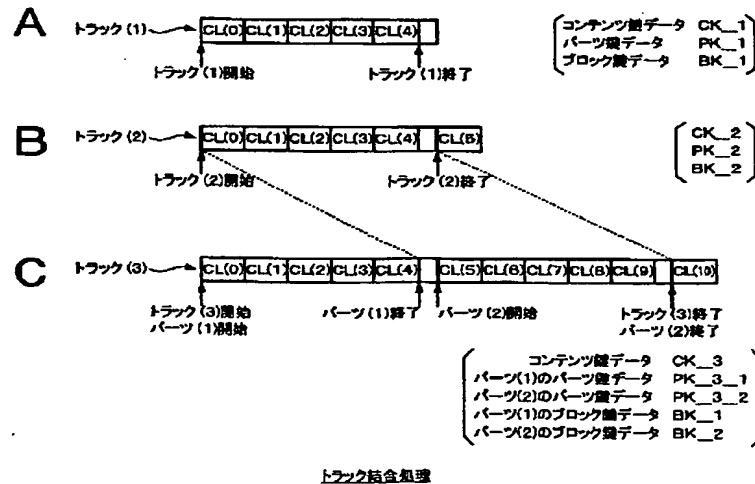


携帯用記憶装置3からの読み出し処理

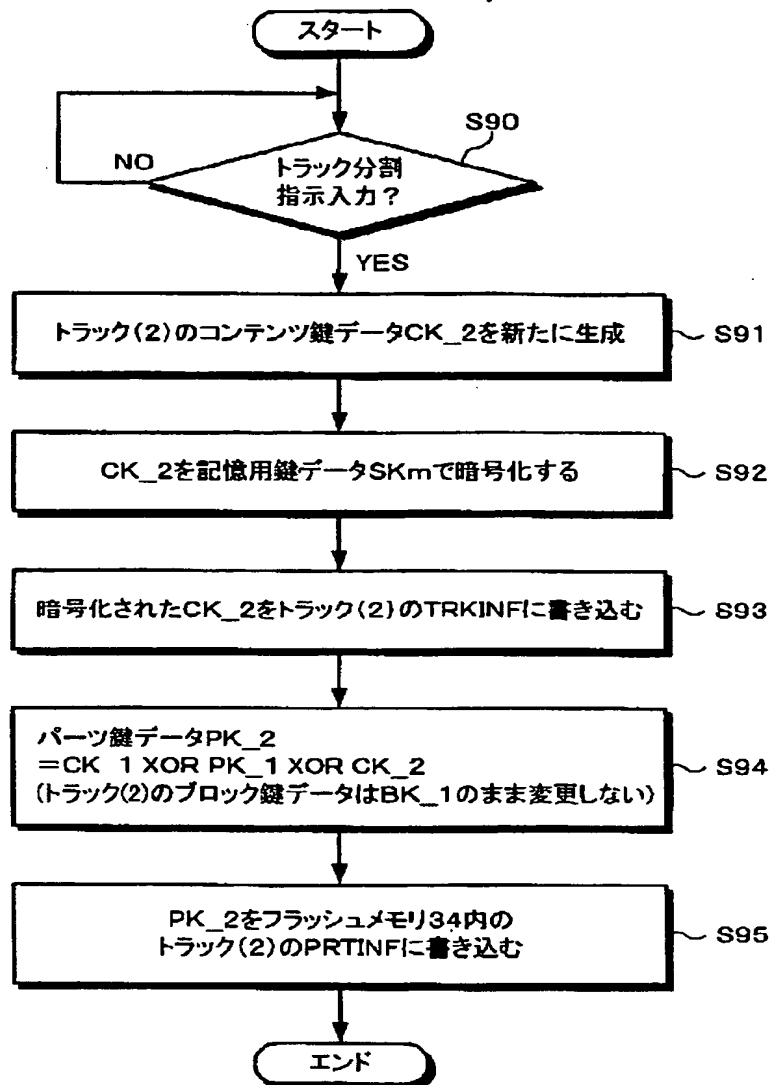
【図27】



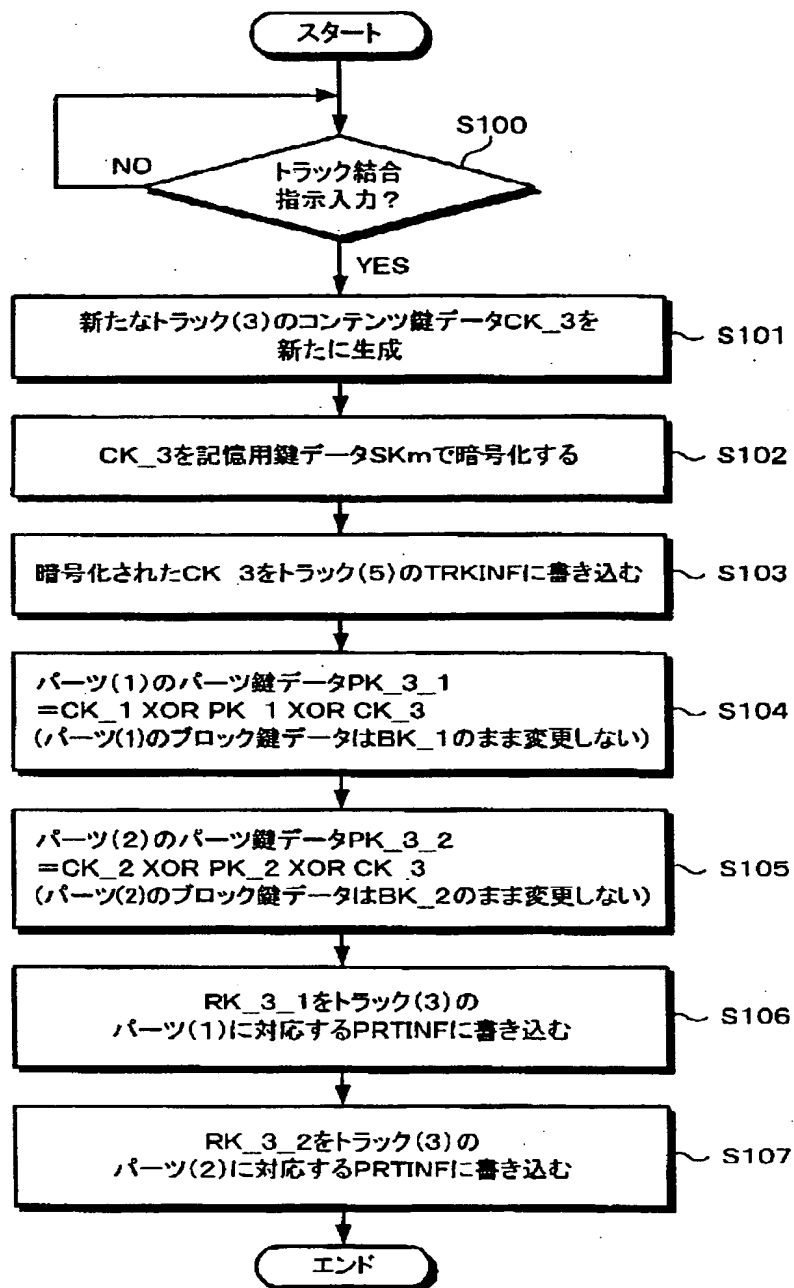
【図30】



【図29】



【図31】



【手続補正書】

【提出日】平成12年4月4日(2000.4.4)

【手続補正1】

【補正対象書類名】明細書

【補正対象項目名】請求項1

【補正方法】変更

【補正内容】

【請求項1】 単数または複数の関連するモジュールか

ら構成されるトラックデータを、当該トラックデータに割り当てられた第1の鍵データおよび上記モジュール毎に割り当てられた第2の鍵データを用いて暗号化して記憶装置に出力するデータ処理装置において、上記モジュール毎に、上記第1の鍵データおよび上記モジュールに割り当てられた上記第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、

上記トラックデータを上記モジュール毎に当該モジュールの上記第3の鍵データに応じて暗号化して上記記憶装置に出力する暗号化手段とを有し、

上記鍵データ算出手段は、上記第1の鍵データに変更があった場合に、上記第3の鍵データを変更しないように、上記モジュール毎に上記第2の鍵データを変更するデータ処理装置。

【手続補正2】

【補正対象書類名】明細書

【補正対象項目名】請求項3

【補正方法】変更

【補正内容】

【請求項3】 請求項1において、

上記モジュールは、単数または複数のサブモジュールを有し、上記サブモジュール毎に第4の鍵データがさらに割り当てられている場合に、上記鍵データ算出手段は、上記サブモジュール毎に、上記第3の鍵データおよび上記サブモジュールに割り当てられた上記第4の鍵データから第5の鍵データを算出し、
上記暗号化手段は、上記トラックデータを上記サブモジュール毎に当該サブモジュールの上記第5の鍵データを用いて暗号化して上記記憶装置に出力するデータ処理装置。

【手続補正3】

【補正対象書類名】明細書

【補正対象項目名】請求項9

【補正方法】変更

【補正内容】

【請求項9】 請求項8において、

上記データ処理装置は、
上記暗号化手段によって暗号化された上記第1の鍵データと上記第2の鍵データとを上記記憶装置に書き込むデータ処理装置。

【手続補正4】

【補正対象書類名】明細書

【補正対象項目名】0005

【補正方法】変更

【補正内容】

【0005】

【課題を解決するための手段】 上述した課題を解決するために、請求項1の発明は、単数または複数の関連するモジュールから構成されるトラックデータを、当該トラックデータに割り当てられた第1の鍵データおよびモジュール毎に割り当てられた第2の鍵データを用いて暗号化して記憶装置に出力するデータ処理装置において、モジュール毎に、第1の鍵データおよびモジュールに割り当てられた第2の鍵データから第3の鍵データを算出する鍵データ算出手段と、トラックデータをモジュール毎に当該モジュールの第3の鍵データに応じて暗号化して

記憶装置に出力する暗号化手段とを有し、鍵データ算出手段は、第1の鍵データに変更があった場合に、第3の鍵データを変更しないように、モジュール毎に第2の鍵データを変更するデータ処理装置である。

【手続補正5】

【補正対象書類名】明細書

【補正対象項目名】0027

【補正方法】変更

【補正内容】

【0027】 図5は、再生管理ファイルの構成を示し、図6が一つ（1曲）のトラックデータファイル（以下においてATRAC3データファイルの用語がさすものもトラックデータファイルと同義である）の構成を示す。
再生管理ファイルは、16KB固定長のファイルである。ATRAC3データファイルは、曲単位でもって、先頭の属性ヘッダと、それに続く実際の暗号化された音楽データとからなる。属性ヘッダも16KB固定長とされ、再生管理ファイルと類似した構成を有する。

【手続補正6】

【補正対象書類名】明細書

【補正対象項目名】0061

【補正方法】変更

【補正内容】

【0061】 LT（1バイト）

意味：再生制限フラグ（ビット7およびビット6）とセキュリティバージョン（ビット5～ビット0）

機能：このトラックに関して制限事項があることを表す

値：ビット7： 0＝制限なし 1＝制限有り

ビット6： 0＝期限内 1＝期限切れ

ビット5～ビット0：セキュリティバージョン0（0以外であれば再生禁止とする）

FN0（2バイト）

意味：ファイル番号

機能：最初に記録された時のトラック番号、且つこの値は、メモ리카ード内の隠し領域に記録されたMAC計算用の値の位置を特定する

値：1から0x190（400）

MG（D）SERIAL-*nnn*（16バイト）

意味：記録機器のセキュリティブロック（制御モジュール43）のシリアル番号

機能：記録機器ごとに全て異なる固有の値

値：0から0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF

CONNUM（4バイト）

意味：コンテンツ累積番号

機能：曲毎に累積されていく固有の値で記録機器のセキュリティブロックによって管理される。2の32乗、42億曲分用意されており、記録した曲の識別に使用する

値：0から0xFFFFFFFFF。

フロントページの続き

(51) Int. Cl.⁷

識別記号

F I

テマコード (参考)

G 1 0 L 19/00

G 1 0 L 9/00

N

G 1 1 B 27/031

H 0 4 L 9/00

6 0 1 Z

H 0 4 L 9/08

6 4 1

9/14

G 1 1 B 27/02

A